

Fault-tolerant Holonomic Quantum Computation in Surface Codes

Yi-Cong Zheng* and Todd A. Brun†

*Ming Hsieh Department of Electrical Engineering,
Center for Quantum Information Science & Technology,
University of Southern California,
Los Angeles, California, 90089*

We show that universal holonomic quantum computation (HQC) can be achieved fault-tolerantly by adiabatically deforming the gapped stabilizer Hamiltonian of the surface code, where quantum information is encoded in the degenerate ground space of the system Hamiltonian. We explicitly propose procedures to perform each logical operation, including logical state initialization, logical state measurement, logical CNOT, state injection and distillation, etc. In particular, adiabatic braiding of different types of holes on the surface leads to a topologically protected, non-Abelian geometric logical CNOT. Throughout the computation, quantum information is protected from both small perturbations and low weight thermal excitations by a constant energy gap, and is independent of the system size. Also the Hamiltonian terms have weight at most four during the whole process. The effect of thermal error propagation is considered during the adiabatic code deformation. With the help of active error correction, this scheme is fault-tolerant, in the sense that the computation time can be arbitrarily long for large enough lattice size. It is shown that the frequency of error correction and the physical resources needed can be greatly reduced by the constant energy gap.

PACS numbers: 03.65.Vf, 03.67.Lx, 03.67.Pp

I. INTRODUCTION

Quantum computers (QCs) provide the means to solve certain problems that cannot be handled classically; yet they are extremely vulnerable to errors during the computation [1]. The threshold theorem indicates that if errors are all local and their rates are below a certain threshold, it is possible to implement large scale quantum computation with arbitrarily small error [2–6] based on active quantum error correction (QEC). However the threshold is difficult to achieve, and tremendous physical resources are required, making QCs very difficult to build in practice.

In addition to protecting QCs by active QEC, much work has been done on providing inherent robustness through the hardware design, such as holonomic quantum computation (HQC) [7], adiabatic quantum computing (AQC) [8, 9], topological quantum computation (TQC) [10–12]. However, these methods all have advantages and disadvantages, which are detailed below. In this paper, we will combine the good features of these architectures and avoid their weakness by proposing the scheme of fault-tolerant HQC in surface codes.

Holonomic QC uses the non-Abelian generalization of Berry phase [13] induced by deforming the Hamiltonian adiabatically and cyclic (closed-loop) to obtain unitary gates in the ground space. These unitary gates depend only on the geometry of the paths in the control manifold. This approach has been shown to be robust against various types of errors during the process [14–16] and could in principle be done

in several different systems [17–19]. Both closed-loop and open-loop HQCs can be compatible with active QEC [20–24], and can achieve fault-tolerant QC. However, for small quantum systems, it is difficult to maintain the degeneracy of the ground space, which is easily broken by even small perturbations, causing unavoidable phase errors.

Another method is to use adiabatic quantum computing (AQC) by slowly changing the Hamiltonian to a special final Hamiltonian, whose ground state encodes the solution of the problem to be solved [8, 9]. This method completely drops the standard circuit model. AQC can suppress thermal noise when the evolution is very slow [25], because of the non-zero energy gap between the ground state and the other excited states. While considerable work has been done in this direction, such as in Ref. [26, 27], a fault-tolerance theorem for AQC is still lacking. Typically, the minimum energy gap of the system scales as an inverse polynomial in the problem size [28, 29], so that the temperature must be arbitrarily low to prevent thermal excitation.

A third method is the beautiful idea of topological quantum computation (TQC) first introduced by Kitaev [10], where excited states of system Hamiltonian behave like particles with exotic statistics, called anyons. By adiabatically braiding anyons around one another in space-time, it induces the unitary operation that depends only on the topology of the anyon world lines. Remarkably, some systems can support non-Abelian anyons, perform universal quantum computation on information encoded in the label space of the anyons [30], while being protected by an energy gap independent of the system size. Unlike HQC, TQC is immune to the effect of small perturbations, since quantum information is stored and processed nonlocally, so that the splitting of the degenerate ground space will decrease exponentially with the system

* yicongzh@usc.edu

† tbrun@usc.edu

size [31]. However, this topological protection does not completely eliminate the need for active error correction. The energy gap can protect information only to a certain extent, and unwanted anyons could be created if the computation time is long enough. Besides, unwanted anyons may be generated during the process of creation, fusion and imperfect adiabatic motion of anyons, and they may not be detectable. One must measure anyon occupations to determine when and where unwanted anyons are created [32], but this is usually difficult in most TQC models (like fractional quantum Hall systems).

On the other hand, a combination of ideas from TQC and QEC gives schemes of active error correction architecture based on topological QEC codes, especially the surface codes [33, 34] and color codes [35], using code deformation [36, 37]. In this approach, one works directly with the quantum error correcting code used in TQC, without introducing a Hamiltonian to protect quantum information with energy gap [38]. In the case of surfaces code, one truncates it by turning off some stabilizer generators in a region to create a hole or defect. Rather than encoding information in the label space of anyons in TQC, each hole can be viewed as an encoded qubit. Via a sequence of measurements, the boundary of holes can be deformed. One can then braid holes by using suitable deformations to perform logical operations between logical qubits associated with the holes. Because of its tolerance of local errors [38], scalable structure and high threshold (0.57%) [39, 40], surface codes have attracted a great deal of attention, and impressive experimental progress in this direction has been made recently with superconducting qubits [41].

In this paper, we try to combine the best features of all the architectures mentioned above, and avoid their weakness. We focus on surface codes with a stabilizer Hamiltonian turned on to form a topological quantum memory [38, 42] on a *single* 2D lattice, to protect quantum information encoded in the degenerate ground space from both thermal errors and perturbations. We explicitly construct all processes needed to do universal holonomic quantum computation (HQC) based on the surface code, by adiabatically deforming this gapped Hamiltonian. By adiabatically braiding different types of holes on the surface, one performs a topologically protected non-Abelian geometric logical CNOT gate. Throughout the entire information processing procedure, including logical state initialization, logical state measurement, logical gates, state injection and distillation, quantum information is protected from local thermal excitations by a constant energy gap, and the weight of the Hamiltonian terms is bounded by 4 during the whole adiabatic code deformation process. To deal with unwanted excitations caused by errors (creation of anyons) during the adiabatic code deformation, we analyze errors propagation, and give conditions when turning off the stabilizer Hamiltonian is needed to do syndrome measurement and error correction. It can be shown that with gap protection the frequency of error correction and the physical resources needed can be greatly reduced. We conclude that the computation procedures are scalable, and that the scheme is fault tolerant.

II. PRELIMINARY

A. Surface Code

A good introduction to the surface code can be found in Refs. [39, 40]. In this section, we follow Ref. [40] and give a brief review to establish our notation. Surface codes can be viewed as a special kind of stabilizer codes defined on a 2D square lattice. In this paper, we implement the surface code on a two-dimensional $L \times L$ lattice, with qubits on the edges of the lattice, as shown in Fig. 1 for $L = 8$. The stabilizer generators of surface codes are two different kinds of operators:

$$X_s = \prod_{i \in s} \sigma_{x_i}, \quad Z_p = \prod_{i \in p} \sigma_{z_i}, \quad (1)$$

that represents vertices (X_s) and plaquette operators (Z_p) on the square lattice.

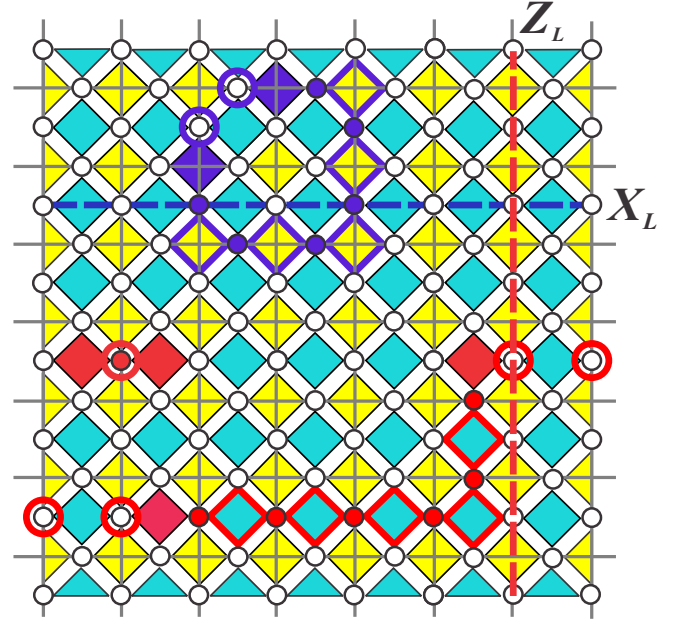


FIG. 1. (Color Online) A surface code based on an 8×8 lattice with 113 physical qubits on the edges. This code contains 1 logical qubit and has distance $d = L = 8$, where d is the distance of the code. The four-body (or three-body) plaquette stabilizer generator (Z_p) and vertex stabilizer generator (X_s) are indicated as cyan and yellow plaquettes, respectively inside the lattice (or on the boundary). A particular choice of logical operators X_L and Z_L is shown. A number of qubits are affected by σ_x (red dots) or σ_z (purple dots) errors, leading to excited Z_p operators (or m anyons) and X_s operators (or e anyons). Measuring these operators yields the positions of the excited vertices and plaquettes but reveals no information about the actual physical errors which cause them. A minimum-weight matching error correction procedure applies σ_x and σ_z to the qubits marked by the larger red and purple circles. While the σ_z errors are annihilated properly (up to a trivial loop of multiplication of Z_p operators), the red pair underneath is connected by a topologically non-trivial path across the surface. This introduces a logical error in the state to be protected.

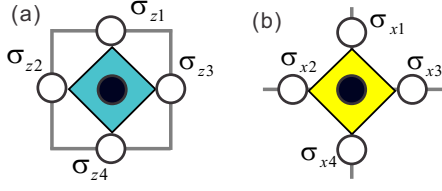
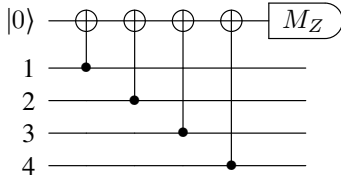


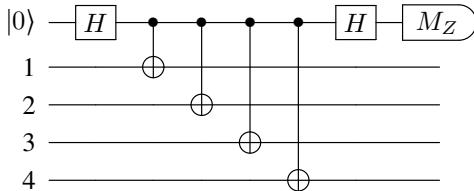
FIG. 2. (Color online) Four-body plaquette operator Z_p (a) and vertex operator X_s (b) as stabilizer generators of surface code inside the lattice. The black dot in the center of the plaquettes are syndrome qubits used to do stabilizer measurement.

Besides stabilizer generators inside the lattice, there are also ones on the boundaries for each lattice. Typically, for each surface code, there are two kinds of boundaries: X boundaries and Z boundaries. X boundaries comprise three-body X_s operators on the boundary of lattice, while Z boundaries comprise three-body Z_p operators, as in the boundaries shown in Fig. 1. In general, a lattice with two X boundaries and two Z boundaries has $2L^2 - 2L + 1$ qubits and $2L^2 - 2L$ stabilizer generators, and encodes 2 degrees of freedom to form a logical qubit. The corresponding logical operators are given by $Z_L = \prod_{k \in l_z} \sigma_{z_k}$ and $X_L = \prod_{k \in l_x} \sigma_{x_k}$ where l_z and l_x are chains of qubits that support σ_z and σ_x operators all the way across the lattice (see Fig. 1 for an example).

Not shown in Fig. 1 are additional *syndrome qubits* for each plaquette and vertex, that enable one to check the sign of the associated stabilizer generator, as shown in Fig. 2. Inside the surface, each syndrome qubit contacts four data qubits and performs four-qubit joint measurement. On the boundaries, each syndrome qubit contacts only three data qubits and performs a three-qubit joint measurement. The corresponding quantum circuit for one stabilizer generator measurement of the Z_p and X_s operators are



and



respectively. The syndrome qubits are always initialized to $|0\rangle$ before the measurement.

If no errors of any kind occur, the code remains in the simultaneous $+1$ eigenstate of all stabilizer generators. We will restrict our attention to σ_x bit-flip errors and σ_z phase-flip errors, since very general noise can be tolerated with just the

ability to correct these two types of error. If σ_x or σ_z errors occur, the value of the stabilizer generators anticommute with errors will be flipped to -1 . Fig. 1 shows the effect of σ_x and σ_z errors on the surface. If we can reliably detect when stabilizer generators become negative, it is possible for us to detect the errors and correct them by finding paths that connect the flipped syndromes of same kind such that the total number of path edges is minimized. Note that σ_x errors can also be matched to X boundaries and σ_z errors can be matched to Z boundaries of the surfaces. An example of decoding failure is also shown in Fig. 1.

However, the syndrome measurement processes are not necessarily perfect. It is possible for the reported measurement outcome to be wrong because of the imperfect CNOT gates and measurement errors. To get around this problem, one needs to keep track of every time the reported eigenvalue of each stabilizer generator changes. Pairs of flipped syndromes are then connected by paths in both space and time, such that total number of edges connected in space-time used to decode the errors is minimal. Polynomial time minimum weight matching algorithms exists [43], and hence this can be done efficiently.

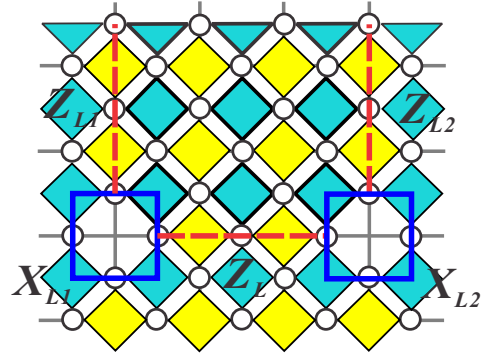


FIG. 3. (Color online). An example of double X -cut qubit, with X_s operators turned off. Each X -cut hole forms a single X -cut logical qubit and there are two kinds of logical operators. Z_{L1} (Z_{L2}) connects left (right) hole with the Z boundary on the top of lattice, while X_{L1} (X_{L2}) are any loops encircling left (right) hole. For double X -cut qubit, there is a more convenient way to define the logical operators is to set $X_L = X_{L1}$ and $Z_L = Z_{L1}Z_{L2}$. Note that Z_L is equivalent to $Z_{L1}Z_{L2}$ up to multiplication by Z_p operators inside the loop and has effect of flip phases for both qubit holes.

For the single logical qubit encoded in surface code, its logical operators X_L and Z_L compose chains of σ_x and σ_z operators crossing the entire lattice. So this way of encoding is not suitable for larger lattices. Besides, no matter how large the lattice, only a single logical qubit can be stored, since the dimension of code space is fixed. A more flexible approach of encoding is to create holes, or defects, inside the lattice to build extra boundaries on lattice. This can be done by turning off one or more of the X_s and Z_p stabilizer generators inside lattice to form a hole. Here, “turn off” means that syndrome measurement is no longer performed for this operator

(see example in Fig. 3) in subsequent computation, so that extra degrees of freedom can be obtained to form a logical qubit. We call the logical qubit obtained this way an X -cut (Z -cut) *single* logical qubit when an X_s (Z_p) stabilizer generator is turned off. For the case of Fig. 3, any chain of σ_z operators connecting this hole to an X -boundary on the top of lattice and any chain of σ_x operators encircling the X -cut hole can be used to manipulate these extra degrees of freedom. We call any such σ_z chain Z_L , and any σ_x ring X_L . If the eigenvalue of X_s is $+1$ (-1) before it is turned off, the logical qubit is initialized to the $|+\rangle$ ($|-\rangle$) state of X -cut single logical qubit, we represent it as $|+_{SL}^X\rangle$ ($|-_{SL}^X\rangle$).

One can go further by making logical operators for qubit not rely on operator chains that reach the boundary of lattice. In particular, we can use a pair of X (Z)-cut holes to form a single logical qubits and manipulate them in a correlated way. This pair of holes are called *double* X (Z)-cut logical qubit. Fig. 3 shows an example of a double X -cut qubit. Four additional degrees of freedom will be added to the lattice when two X_s are turned off, which can be represented as:

$$\begin{aligned} &|+_{SL}^X\rangle_1|+_{SL}^X\rangle_2, \quad |-_{SL}^X\rangle_1|-_{SL}^X\rangle_2, \\ &|-_{SL}^X\rangle_1|+_{SL}^X\rangle_2, \quad |+_{SL}^X\rangle_1|-_{SL}^X\rangle_2, \end{aligned} \quad (2)$$

where 1 denotes the single X -cut qubit on the left and 2 denotes the one on the right. Each single X -cut qubit can be manipulated by defining X_{L_1} and Z_{L_1} for the left X -cut hole and X_{L_2} and Z_{L_2} for the right X -cut hole. The effect of each logical operator pair is:

$$X_{L_j}|\pm_{SL}^X\rangle_j = \pm|\pm_{SL}^X\rangle_j, \quad Z_{L_j}|\pm_{SL}^X\rangle_j = |\mp_{SL}^X\rangle_j. \quad (3)$$

Manipulating the two qubit holes of a double cut logical qubit in a correlated way can greatly simplify the forms of logical operators and increase the number of logical qubits encoded on a single lattice. We can define the $|+\rangle$ and $|-\rangle$ states for double X -cut logical qubits as:

$$|+_{DL}^X\rangle = |+_{SL}^X\rangle_1|+_{SL}^X\rangle_2, \quad |-_{DL}^X\rangle = |-_{SL}^X\rangle_1|-_{SL}^X\rangle_2. \quad (4)$$

A chain of σ_z operators connecting the two holes is then used as the definition of the Z_L operator for double X -cut qubit, as shown in Fig. 3. The X_L operator can be defined as any ring of σ_x operators around either hole, as can be seen from Eq. (3). We can then find the $|0\rangle$ state for the double X -cut qubit:

$$\begin{aligned} |0_{DL}^X\rangle &= \frac{1}{\sqrt{2}}(|+_{SL}^X\rangle_1|+_{SL}^X\rangle_2 + |-_{SL}^X\rangle_1|-_{SL}^X\rangle_2), \\ |1_{DL}^X\rangle &= \frac{1}{\sqrt{2}}(|+_{SL}^X\rangle_1|+_{SL}^X\rangle_2 - |-_{SL}^X\rangle_1|-_{SL}^X\rangle_2). \end{aligned} \quad (5)$$

Similarly, the $|0\rangle$ and $|1\rangle$ states of double Z -cut qubits can be defined as:

$$|0_{DL}^Z\rangle = |0_{SL}^Z\rangle_1|0_{SL}^Z\rangle_2, \quad |1_{DL}^Z\rangle = |1_{SL}^Z\rangle_1|1_{SL}^Z\rangle_2 \quad (6)$$

and the corresponding $|+\rangle$ and $|-\rangle$ states of double Z -cut

qubits are

$$\begin{aligned} |+_{DL}^Z\rangle &= \frac{1}{\sqrt{2}}(|0_{SL}^Z\rangle_1|0_{SL}^Z\rangle_2 + |1_{SL}^Z\rangle_1|1_{SL}^Z\rangle_2), \\ |-_{DL}^Z\rangle &= \frac{1}{\sqrt{2}}(|0_{SL}^Z\rangle_1|0_{SL}^Z\rangle_2 - |1_{SL}^Z\rangle_1|1_{SL}^Z\rangle_2). \end{aligned} \quad (7)$$

Note that for the logical qubits described here, the distance of the codes is bounded by 4, no matter how far two holes are separated, because the perimeter of hole created by turning off one stabilizer generator is limited by 4 physical qubits. The error correction ability can be significantly improved if we increase both the size and spacing of the two holes, as this will increase the number of physical qubits involved in Z_L and X_L . The details of making larger holes for logical qubits will be discussed in Sec. IV B.

B. Holonomic Quantum Computation

Consider a Hamiltonian family $\{H_\lambda\}$ on an N -dimensional Hilbert space. The point λ , parametrizing the Hamiltonian, is an element of a manifold \mathcal{M} called the control manifold, and the local coordinates of λ are denoted by λ^i ($1 \leq i \leq \dim \mathcal{M}$). Assume there are only a fixed number of eigenvalues $\{\varepsilon_k(\lambda)\}$ and suppose the n th eigenvalue $\varepsilon_n(\lambda)$ is K_n -fold degenerate for any λ . The degenerate subspace at λ is denoted by $\mathcal{H}_n(\lambda)$. The orthonormal basis vectors of $\mathcal{H}_n(\lambda)$ are denoted by $\{|\phi_\alpha^n; \lambda\rangle\}$, satisfying

$$H_\lambda|\phi_\alpha^n; \lambda\rangle = \varepsilon_n(\lambda)|\phi_\alpha^n; \lambda\rangle, \quad (8)$$

and

$$\langle\phi_\alpha^n; \lambda|\phi_\beta^m; \lambda\rangle = \delta_{nm}\delta_{\alpha\beta}. \quad (9)$$

Assume the parameter λ is changed adiabatically, which means that

$$(\varepsilon_n(\lambda(t)) - \varepsilon_{n'}(\lambda(t)))T \gg 1 \quad (10)$$

is satisfied for $n \neq n'$ during $0 \leq t \leq T$. Suppose the initial state at $t = 0$ is an eigenstate $|\psi^n(0)\rangle = |\phi_\alpha^n; \lambda(0)\rangle$. The Schrödinger equation is

$$i\frac{d}{dt}|\psi^n(t)\rangle = H(\lambda(t))|\psi^n(t)\rangle, \quad (11)$$

whose solution will have the form

$$|\psi^n(t)\rangle = \sum_{\beta=1}^{K_n} |\phi_\beta^n; \lambda(t)\rangle U_{\beta\alpha}(t). \quad (12)$$

where we have used the adiabatic approximation from Eq. (10). Substituting Eq. (12) into Eq. (11), one finds that $U_{\beta\alpha}$ satisfies

$$\begin{aligned} \dot{U}_{\beta\alpha}(t) &= -i\varepsilon_n(\lambda(t))U_{\beta\alpha}(t) \\ &\quad - \sum_{\mu} \langle\phi_\beta^n; \lambda(t)|\frac{d}{dt}|\phi_\mu^n; \lambda(t)\rangle U_{\mu\alpha}(t). \end{aligned} \quad (13)$$

The solution can be expressed as

$$U(t) = \exp\left(-i \int_0^t \varepsilon_n(\lambda(s)) ds\right) \times \mathcal{T} \exp\left(-\int_0^t A^n(\tau) d\tau\right), \quad (14)$$

where \mathcal{T} is the time-ordering operator and

$$A_{\beta\alpha}^n(t) = \langle \phi_\beta^n; \lambda(t) | \frac{d}{dt} | \phi_\alpha^n; \lambda(t) \rangle \quad (15)$$

is the Wilczek-Zee (WZ) connection [13]. Define the connection

$$\mathcal{A}_{i,\beta\alpha}^n(t) = \langle \phi_\beta^n; \lambda(t) | \frac{\partial}{\partial \lambda^i} | \phi_\alpha^n; \lambda(t) \rangle, \quad (16)$$

through which $U(t)$ can be expressed as

$$U(t) = \exp\left(-i \int_0^t \varepsilon_n(\lambda(s)) ds\right) \times \mathcal{P} \exp\left(-\int_{\lambda(0)}^{\lambda(t)} \sum_i \mathcal{A}_i^n d\lambda^i\right), \quad (17)$$

where \mathcal{P} is the path-ordering operator. Eq. (17) is a general description of both open loop and closed loop adiabatic state evolution. Both are useful for our scheme as will be shown in Sec. III and Sec. IV. In particular, suppose the path $\lambda(t)$ is a loop λ in \mathcal{M} such that $\lambda(0) = \lambda(T) = \lambda_0$ (closed loop). Then after transporting through λ , states are transformed to

$$|\psi^n(T)\rangle = \sum_{\beta=1}^{K_n} |\psi_\beta^n(0)\rangle U_{\beta\alpha}(T). \quad (18)$$

The unitary matrix

$$\Gamma_\lambda = \mathcal{P} \exp\left(-\oint_\lambda \sum_i \mathcal{A}_i^n d\lambda^i\right) \quad (19)$$

is called the holonomy associated with the loop $\lambda(t)$. Γ_λ is a purely geometric object, and is independent of the parametrization of the path. Note that for a given Γ_λ , there exist infinitely many paths λ . One of the main objects of the paper to find the proper path in \mathcal{M} that will give us the desired state transformation in the code space of the surface code under adiabatic transformation of the stabilizer Hamiltonian. A geometric formulation of the holonomic problem, which gives an alternative description as shown in Refs. [44, 45], is also given in Appendix A, which is useful in improving the results of the next section.

III. SKETCH OF THE SCHEME

In this scheme, we always regard all physical qubits on the lattice as a single big stabilizer code. We assume that the qubits independently and weakly interact with a thermal bath

in the Markovian approximation. The corresponding thermal errors are local and low-weight during a certain period of evolution. Those low-weight thermal excitations will cause transitions from the ground space to excited spaces. Their rate should decrease as $\delta_{\text{thermal}} \sim \exp(-c\beta\Delta_{\min})$, where Δ_{\min} is the minimum spectral gap of the system, β is the inverse of temperature, and c is a constant depending on the coupling strength between system and thermal bath [46]. This is true even when the Hamiltonian is not static and changes slowly, so long as the system is weakly coupled to the thermal bath [25]. The goal is to do the whole quantum computation fault-tolerantly, while the code space is protected by an energy gap of the stabilizer Hamiltonian that exponentially suppresses errors at low temperature throughout the information processing procedure.

To analyze the error performance of the architecture, we must first define a fault-tolerant procedure:

Definition 1. A procedure is fault-tolerant if it has the property that if only one component (or more generally, a small number of components) in the procedure fails, the errors produced by this failure are not transformed into an uncorrectable error by the procedure, before error correction is applied.

With this definition, the fault-tolerance of a procedure can be regarded as a property of the procedure itself regardless of the error model of the system. Before we go deeper, we must impose some requirements to follow in the rest of the paper:

1. Procedures like logical state preparation, logical state measurement, encoded gate operations, state injection and state distillation should be done when the system Hamiltonian is “turned on”, so that a constant energy gap protects the information and the error rate for each procedure is low.
2. All procedures should be done fault-tolerantly according to Def. 1, whether adiabatic or not.
3. Syndrome measurements and error correction should be done before uncorrectable errors happen.
4. Syndrome measurements and error correction should be done as seldom as possible, since they are in general not compatible with the system Hamiltonian and we must turn off the Hamiltonian before doing them. Besides, the syndrome measurement procedure itself is quite expensive. The frequency of error correction is expected to be low if all procedures are gap-protected.
5. A threshold theorem should exist, in the sense that if the error rate is below the threshold, the computation can be made arbitrarily long by suitably increasing the lattice size.
6. It is possible to measure σ_x and σ_z of single physical qubits in certain circumstances even when the Hamiltonian is turned on.
7. Maximum weight of the Hamiltonian terms should be low, and the Hamiltonian should be geometrically local.

8. All procedures should be done in a single lattice.

Requirements 1 – 4 are crucial to our main objective of reducing the physical resources and 5 guarantees that arbitrarily large-scale computation can be done. Requirement 6 is physically reasonable, and we will see its importance in Sec. IV. Requirement 7 comes from the fact that in real experiments, high weight and nonlocal Hamiltonians are difficult or impossible. Requirement 8 is technical rather than fundamental, since it simplifies the the computation architecture.

A. Adiabatic processes

In most cases, adiabatic processes can be used to simultaneously fulfill most of the requirements above. For the purpose of encoding and measuring logical qubits, we will show that these can be done by open-loop adiabatic processes (for logical measurement, we also need qubit measurement), while the logical CNOT can be done by a closed-loop adiabatic processes to get a holonomy on the code space. Both such processes can be described by Eq. (17). In this and the following sections, we will focus on a special kind of adiabatic evolution that turns out to be particularly useful. In addition, we will discuss how it can be used to analyze propagation of potential errors and parallelism of the processes.

Assume the total number of qubits on the lattice is $n = 2L^2 - 2L + 1$, so the dimension of the Hilbert space is $N = 2^n$. The number of logical qubits in our scheme may change over time, since we can create defects on the lattice to create logical qubits. However, we assume that when an adiabatic process is applied, the dimension of code space is fixed. This can be realized by isospectral deformation of the Hamiltonian. Denote the number of logical qubits encoded in the ground space by k . Assume that at time t_0 , the initial Hamiltonian can be written as

$$H(t_0) = - \sum_{j=1}^{n-k} JS_j, \quad (20)$$

where the $\{S_j\}$ are a set of stabilizer generators of the surface code at time t_0 and that $\{S_j\}$ forms the stabilizer group \mathcal{S} . Consider the following way to adiabatically deform the Hamiltonian isospectrally:

$$\begin{aligned} H(t) &= - \sum_{j=1}^{n-k} JS_j(t) \\ &= - \sum_{j=1}^{n-k} JU(t, t_0)S_j(t_0)U^\dagger(t, t_0), \end{aligned} \quad (21)$$

with $S_j(t) = U(t, t_0)S_j(t_0)U^\dagger(t, t_0)$ and $[S_i(t), S_j(t)] = 0$ for all i, j . The $\{S_j(t)\}$ can be viewed as a set of generators of an Abelian group, like the stabilizer group. The Hamiltonian also has a spectral decomposition:

$$H(t) = \sum_{\mathbf{s}} \varepsilon_{\mathbf{s}} P_{\mathbf{s}}(t). \quad (22)$$

Here, the $\{P_{\mathbf{s}}(t)\}$ are projectors onto the simultaneous eigenspaces of all the $S_j(t)$, with eigenvalues:

$$\varepsilon_{\mathbf{s}} = -J \sum_j s_j, \quad (23)$$

where the labels $s_j = \pm 1$ form a vector:

$$\mathbf{s} = \{s_1, s_2, \dots, s_{n-k}\}. \quad (24)$$

The ground space evolves with the system Hamiltonian. This defines a time-dependent code space \mathcal{C}_t . Let $P_0(t) = U(t, t_0)P_0(t_0)U^\dagger(t, t_0)$ be the projector onto the ground space of $H(t)$, such that $s_j = 1$ for all j . We emphasize that $U(t + \tau, t)$ should be chosen such that

$$\left[\frac{\partial}{\partial \tau} U(t + \tau, t) \Big|_{\tau=0}, P_{\mathbf{s}}(t) \right] \neq 0 \quad \text{for all } \mathbf{s}, \quad (25)$$

for any time t , so that the deformation procedure is nontrivial for all eigenspaces. In other word, $U(t + \tau, t)$ should not belong to the isotropy group of $P_{\mathbf{s}}(t)$ for small values of τ .

The adiabatic condition must hold for each eigenspace $P_{\mathbf{s}}$, so that each eigenspace undergoes nontrivial evolution under the adiabatic process, in case an error excites the system to $P_{\mathbf{s}}$ during the process. The standard adiabatic condition [47] for any eigenspace $\{P_{\mathbf{s}_\alpha}\}$ can be reformulated as:

$$\frac{\| P_{\mathbf{s}_\alpha}(t) \frac{\partial}{\partial t} H(t) P_{\mathbf{s}_\beta}(t) \|_1}{K(\varepsilon_{\mathbf{s}_\alpha}(t) - \varepsilon_{\mathbf{s}_\beta}(t))^2} \approx 0, \quad \text{for any } \alpha \neq \beta. \quad (26)$$

Here, K is the degeneracy of each $P_{\mathbf{s}}$. This must hold for all $t \in [t_0, t_p]$, where $\| \cdot \|_1$ is the trace norm ($\| A \|_1 = \text{Tr} \sqrt{A^\dagger A}$). It is very likely that for a Hamiltonian of the form Eq. (21), several $P_{\mathbf{s}}(t)$'s will share the same eigenenergy, so that the adiabatic condition cannot be directly satisfied. Fortunately, for the surface code, we will show later that there is a natural way to cope with this problem, so that each $P_{\mathbf{s}}(t)$ can satisfy the adiabatic condition during the adiabatic code deformation.

As shown in Ref. [24], a closed loop adiabatic logical gate operation can be built from a fault-tolerant circuit of the corresponding stabilizer code. However, for the surface code, we in general don't know the exact fault-tolerant circuit for encoded gate operations. Moreover, we wish to do encoding and logical state measurement with gap protection, so the result in Ref. [24] cannot be applied here directly. Instead, in this paper, we consider a special kind of quantum circuit \mathcal{G} composed of a sequence of gate operations $\{g_1, g_2, \dots, g_p\}$ giving the unitary operation $\Omega_p = \prod_{l=1}^p g_l$. Here, $g_l = \exp(i \frac{\pi}{4} Q_l)$ for some Hermitian operator $Q_q \in G_n$, where G_n is the Pauli group acting on n qubits. For simplicity, when we talk about a "circuit" in the rest of paper, we means the circuit of this type. We divide the information processing time $[t_0, t_p]$ into p small steps and represent the q th time segment as $[t_{q-1}, t_q]$. Now, set the unitary operator

$$U_q(t, t_{q-1}) = \exp(i f_q(t) Q_q), \quad (27)$$

for $t \in [t_{q-1}, t_q]$ and let $f_q : [t_{q-1}, t_q] \rightarrow [0, \pi/4]$ be a monotonic smooth function with boundary conditions $f_q(t_{q-1}) = 0$

and $f_q(t_q) = \pi/4$. For each time segment $[t_{q-1}, t_q]$, we adiabatically deform the Hamiltonian:

$$H(t, t_{q-1}) = U_q(t, t_{q-1})H(t_{q-1})U_q^\dagger(t, t_{q-1}), \quad t \in [t_{q-1}, t_q] \quad (28)$$

and assume $[Q_q, H(t_{q-1})] \neq 0$ so that Eq. (25) is satisfied. A state in the ground space will evolve as described by the following lemma:

Lemma 1. (State Evolution) Consider a circuit composed of gates $\{g_q\}$ and an initial state $|\psi(t_0)\rangle \in \mathcal{C}(t_0)$, with $H(t_0) = -\sum_j JS_j$. We apply a sequence of Hamiltonian deformations as in Eq. (28), for $1 \leq q \leq p$. Then, under the adiabatic approximation, the final state will be:

$$\begin{aligned} |\psi(t_p)\rangle &= e^{-i\varepsilon_0(t_p-t_0)} \left(\prod_{l=1}^p g_l \right) |\psi(t_0)\rangle \\ &= e^{-i\varepsilon_0(t_p-t_0)} \Omega_p |\psi(t_0)\rangle. \end{aligned} \quad (29)$$

Proof. See Appendix B 1. \square

In the case of a many-body system like the surface code, it is difficult to follow the change of the state in code space since it is hard to represent the state. One normally uses the stabilizer formalism (Heisenberg picture) to track the change of the logical Z_L and X_L operators during the process. The following theorem is a direct consequence of Lemma 1:

Theorem 1. Suppose the initial state $|\psi(t_0)\rangle$ is in the code space of a stabilizer code with generators $\{S_j\}$ and logical operators $\{X_L^i, Z_L^i\}$, and that $H(t_0) = -\sum_j JS_j$. Under the adiabatic Hamiltonian deformation described in Eq. (28) for $1 \leq q \leq p$, the logical operators will map to $X_L^i \rightarrow \Omega_p X_L^i \Omega_p^\dagger$, $Z_L^i \rightarrow \Omega_p Z_L^i \Omega_p^\dagger$, and the system Hamiltonian will become $H(t_p) = -\sum_j JS'_j = -\sum_j J\Omega_p S_j \Omega_p^\dagger$.

If the process is cyclic for the ground space, which means $\Omega_p P_0(t_0) \Omega_p^\dagger = P_0(t_0)$, then Ω_p can be viewed as an encoded gate operation, and we have following conclusion:

Corollary 1. If $\Omega_p \in N(\mathcal{S}) \setminus \mathcal{S}$, where $N(\mathcal{S})$ is the normalizer of \mathcal{S} in $U(N)$, then Ω_p is a closed-loop holonomic operation under the adiabatic process.

Remark 1. These results build a relationship between the special kind of circuits \mathcal{G} we are interested in and the corresponding adiabatic process. If we can find a circuit in \mathcal{G} giving a particular unitary, then we can translate it to an adiabatic process. However, in general, the weight of the Hamiltonian terms changes with time, and it is quite possible that during the adiabatic process, the Hamiltonian terms will become both nonlocal and high weight. Fortunately, as we will see, in the case of surface codes this can be avoided.

B. Error propagation

Although in the process described by Eq. (21), the ground space is protected by a constant energy gap $2J$, the lifetime

is about $e^{2c\beta J}$ in the presence of a thermal bath. This lifetime doesn't grow with the lattice size L , so the thermal gap does not guarantee fault-tolerance. We still need to do active error correction to make the computation time arbitrarily long. We must analyze how an error caused by thermal excitation will propagate during the adiabatic process to choose the proper circuit from \mathcal{G} and design the subsequent error correction procedure.

Without loss of generality, we assume that an error E_{t_q} happens at time t_q ($q \leq l$). Since any error operator E_{t_q} on an n -qubit system can be decomposed into a sum of Pauli operators $E_{t_q} = \sum_\alpha c_\alpha F_\alpha$, it is sufficient to analyze Pauli errors. We have following lemma:

Lemma 2. (Error Propagation) If an error $E_{t_q} = \sum_\alpha c_\alpha F_\alpha^q$ ($F_\alpha^q \in G_n$) happens at time t_q in the procedure described by Eq. (28), and there is an odd number of stabilizer generators $S_j(t_r)$ such that $[Q_r, S_j(t_{r-1})] \neq 0$ for all times $1 \leq r \leq p$, then

$$|\psi(t_p)\rangle = \sum_\alpha c_\alpha e^{-i\varepsilon_\alpha(t_p-t_q)} F_\alpha^{pq} \left(\prod_{l=1}^p g_l \right) |\psi(t_0)\rangle \quad (30)$$

where $F_\alpha^{pq} = \mathcal{W}^{pq} F_\alpha^q (\mathcal{W}^{pq})^\dagger$ with $\mathcal{W}^{pq} = \prod_{l=q+1}^p g_l$.

Proof. See Appendix B 2. \square

Lemma 2 gives the condition that the error will just propagate to some other error under the expected unitary evolution. The condition that at each step r the number of $S_j(t_{r-1})$ such that $[Q_r, S_j(t_{r-1})] \neq 0$ should be odd is crucial. In general, an error will excite the ground space to another eigenspace P_{s_α} , which will usually share the same energy with some other eigenspaces, so that the adiabatic condition will not hold. This condition guarantees that even when this is the case, the degenerate eigenspaces will still satisfy the adiabatic condition Eq. (26) and adiabatic evolution will not fail.

Also, note that if Ω_p is a logical gate operator, although for the ground space P_0 the process is a cyclic evolution, e.g., $\Omega_p P_0(t_0) \Omega_p^\dagger = P_0(t_0)$, this is not true for the other eigenspaces. In general, $\Omega_p P_s(t_0) \Omega_p^\dagger \neq P_s(t_0)$ for $s \neq 0$. This means that after an error excites the ground space P_0 to P_s , the adiabatic process becomes open loop for P_s .

C. Parallelism of adiabatic operation

The method described in the previous sections is basically a serial operation, meaning that we need to adiabatically deform the Hamiltonian according to the gates in the circuit \mathcal{G} step by step. However, for a large scale QC on a lattice (not only the surface code), we expect that many operations can be done in parallel, so that operations which commute with each other can be done simultaneously. Here we give the condition for those operations to parallelize.

Lemma 3. (Parallelism) Suppose that at time t_q , $|\psi(t_q)\rangle$ is in the ground space $\mathcal{C}(t_q)$. Define $\mathcal{C}_{Q_r} = \{j \mid 1 \leq j \leq n-k, \{S_j(t_q), Q_r\} = 0\}$. Suppose the set of operators $\mathcal{P}_q = \{Q_r \mid q+1 \leq r \leq q+M\}$ satisfies the following conditions:

1. $[Q_r, Q_m] = 0$, for any $Q_r, Q_m \in \mathcal{P}_q$,
2. $\mathcal{C}_{Q_r} \cap \mathcal{C}_{Q_m} = \emptyset$ for any $Q_r, Q_m \in \mathcal{P}_q$,
3. $|\mathcal{C}_{Q_r}|$ is odd for all $Q_r \in \mathcal{P}_q$.

and set $U_{q+1}(t, t_q) = \prod_{r=q+1}^{q+M} \exp(iff(t)Q_r)$ with $f(t) = f_{q+1}(t)$ for $t \in [t_q, t_{q+1}]$. Assume the Hamiltonian changes adiabatically as $H(t) = U_{q+1}(t, t_q)H(t_q)U_{q+1}^\dagger(t, t_q)$. Then we have:

1. The state at time t_{q+1} will be:

$$|\psi(t_{q+1})\rangle = e^{-i\varepsilon_0(t_{q+1}-t_q)} \left(\prod_{l=q+1}^{q+M} g_l \right) |\psi(t_q)\rangle. \quad (31)$$

2. If an error $E_{t_q} = \sum_\alpha c_\alpha F_\alpha^q$ ($F_\alpha^q \in G_n$) occurs at time t_q , then the state at time t_{q+1} will be:

$$|\psi(t_{q+1})\rangle = \sum_\alpha c_\alpha e^{-i\varepsilon_\alpha(t_{q+1}-t_q)} F_\alpha^{q+1,q} \left(\prod_{l=q+1}^{q+M} g_l \right) |\psi(t_q)\rangle, \quad (32)$$

where $F_\alpha^{q+1,q} = \mathcal{U}^{q+1,q} F_\alpha^q (\mathcal{U}^{q+1,q})^\dagger$ with $\mathcal{U}^{q+1,q} = \prod_{r=q+1}^{q+M} g_r$.

Proof. See Appendix B 3. \square

Lemma 3 suggests that it is possible to do M steps of the adiabatic transformation described in Lemma 1 in one step, and gives the conditions for the adiabatic evolution to still be valid when errors occur. This property is extremely important. Since we need to apply our scheme to surface codes of large size, operations applied simultaneously on different parts of the surface can greatly improve the efficiency of computation.

IV. HQC IN SURFACE CODES

We are ready to show how to do QC fault-tolerantly by adiabatically deforming the stabilizer Hamiltonian of the surface code. As mentioned in the previous section, our goal is that all the procedures, including state preparation, ancilla preparation, logical gate operations and logical state measurements, be implemented fault-tolerantly with constant energy gap protection. In the next few subsections, we discuss how to construct these procedures, and discuss error propagation and error detection in detail.

State measurement is a special case worth more discussion here. At the end in the computation, when we want to read all of the data in the logical qubits, we can just turn off the Hamiltonian and measure everything. However, during the computation, when the stabilizer Hamiltonian exists, we still may need to measure logical qubits from time to time, so that actions conditioned on those classical measurement outcomes of logical qubit can be applied. We must put some restrictions on the kinds of measurements we can do that are compatible

with the existence of the stabilizer Hamiltonian. The first requirement is that the observable \mathcal{O} we want to measure should commute with the Hamiltonian:

$$[H, \mathcal{O}] = 0. \quad (33)$$

This requirement guarantees that if a state encoding quantum information is in one of the eigenspaces P_s before the measurement, then after the projective measurement, the state will still be in P_s . If Eq. (33) is not satisfied, the measurement will lead to excitations out of the eigenspace. The second requirement is that the observable should be geometrically local, so that the measurement procedure will not introduce non-local interactions. Note that when the Hamiltonian is turned on, we do not do X_s or Z_p stabilizer measurements even though they commute with the system Hamiltonian and are local. The reason for this is that to projectively measure these many-body observables, we would need to introduce CNOT gates and syndrome qubits, which are not compatible with the system Hamiltonian. So in our scheme, syndrome measurements are always done when the system Hamiltonian is turned off. However, as stated in requirement 6 in the previous section, we do allow single physical qubit measurements as long as they commute with the system Hamiltonian.

Errors can happen during the single qubit measurement process. There are two kinds of measurement errors. The first kind is that, instead of an ideal measurement, some quantum process occurs during the measurement process which is equivalent to one of the following circuits:

$$|\psi\rangle \rightarrow \left[\sigma_x \right] \left[M_Z \right] \quad |\psi\rangle \rightarrow \left[\sigma_z \right] \left[M_X \right]$$

for σ_z measurement and σ_x measurement, respectively. The second kind of error can be regarded as a software error: even though the measurement is perfect, some classical noise corrupts the measurement result and we get the wrong outcome. This can be modeled by the circuits

$$|\psi\rangle \rightarrow \left[M_Z \right] = X \quad |\psi\rangle \rightarrow \left[M_X \right] = X$$

In this paper, we assume we can completely overcome errors of the second kind, and focus only on the first kind of errors.

Finally, note that in the process of computation, we are frequently required to do logical X_L and logical Z_L gates. We do not necessarily implement these gates physically; rather, we can simply keep a record of it, and apply X_L and Z_L to that logical qubit in “software”, as described in Secs. IX and XVI.A of Ref. [40].

A. Creation of $|+\rangle$ ($|0\rangle$) state for X (Z)-cut double qubit

Before computation begins, we assume the system is already prepared with the eigenvalues of all stabilizer generators equal to +1. This can be done by several methods. One of them is preparing all qubits in the $|0\rangle$ state and then measuring

all X_s stabilizer generators and resetting their eigenvalues to +1. After that, we turn on the stabilizer Hamiltonian:

$$H(t_0) = -J \sum_i X_{s_i} - J \sum_j Z_{p_j}. \quad (34)$$

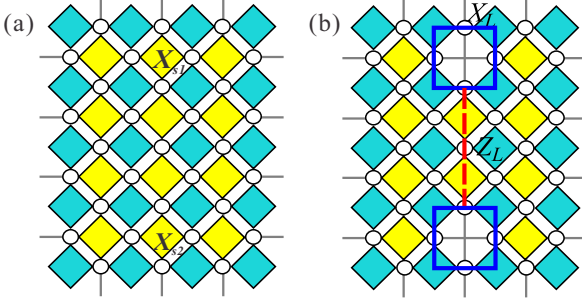


FIG. 4. (Color online) Creation of $|+\rangle$ for X -cut double qubit. System Hamiltonians before and after are shown in (a) and (b) respectively. Colored squares indicate that the corresponding X_s (yellow) and Z_p (cyan) stabilizer generators are turned on.

There are two types of initialization procedures. The first is the creation of a $|+\rangle$ ($|0\rangle$) state for a X (Z)-cut and second is the creation of a $|+\rangle$ ($|0\rangle$) state for Z (X)-cut qubit. Here, we give an example of preparing a $|+\rangle$ state for X -cut double logical qubit; the Z -cut case is similar. We will see that if we can do the first type of preparation fault-tolerantly, we can do the second type fault-tolerantly as well, as will be shown in Sec. IV D. Suppose initially the state of the system is shown in panel (a) of Fig. 4 with a fully stabilized array, and the stabilizer Hamiltonian terms in this area are all turned on. Turning off the X_{s_1} and X_{s_2} terms and makes the Hamiltonian:

$$H(t_1) = -J \sum_{i \neq 1,2} X_{s_i} - J \sum_j Z_{p_j}. \quad (35)$$

This will make the state $|+_{DL}^X\rangle = |+_{SL}^X\rangle_1 |+_{SL}^X\rangle_2$. This process can be done either adiabatically or instantaneously. If errors occur, they will leave nonzero syndromes for future correction, and no errors will be propagated when the X_{s_1} and X_{s_2} terms are turned off.

We can see that the distance for σ_x errors is restricted by 4, no matter how far the pair of holes are separated. To increase the error protection ability of σ_x errors, we need to enlarge the size of the holes. We will describe in detail the adiabatic procedure to enlarge the holes with gap protection in Sec. IV B.

Also note that all state preparations of this type are done right after the initialization of the whole surface, such that X_{s_1} and X_{s_2} are known to be +1 for certain. During the computation, X_{s_1} and X_{s_2} can be flipped to -1 before they are turned off, and we have no way to know their values except by doing syndrome measurement, which we try to avoid. So all qubits needed in the computation are prepared at the beginning.

B. Enlarging the hole

After holes are created, we need to enlarge the size of the hole to improve the ability to correct σ_z (σ_x) errors for Z (X)-cut double qubits. In this section, we will show how to enlarge the hole adiabatically with gap protection. First, we will assume that no error occurs on any qubits during the process. Then we will analyze how errors propagate, and the fault-tolerance of the process. Since this is the first example where we apply the results of Sec. III, we will follow the state transformations based on stabilizer formalism in detail.

1. Scheme

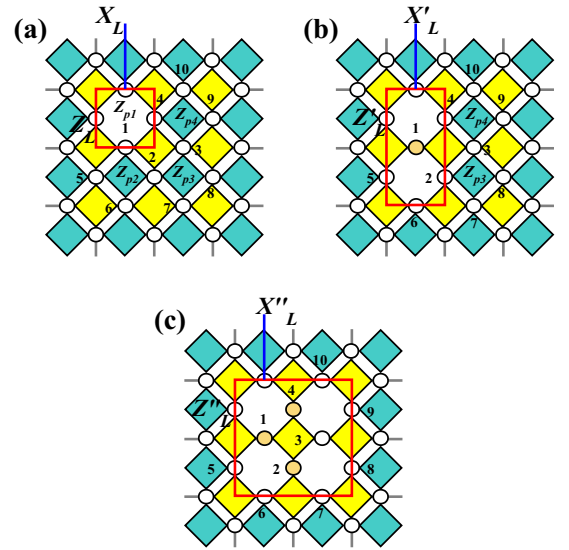


FIG. 5. (Color online) Enlarging a hole of an X -cut logical double qubit adiabatically. Colored squares indicate that the corresponding X_s (yellow) and Z_p (cyan) stabilizer generators are turned on. The yellow qubits in (b) and (c) indicate that σ_x for that qubit is turned on in the Hamiltonian. Adiabatic evolution between (a) and (b) maps X_L to X'_L and Z_L to Z'_L . Similarly, adiabatic evolution between (b) and (c) maps X'_L to X''_L and Z'_L to Z''_L .

Consider the case of a Z -cut qubit, the situation for X -cut qubits is similar. Right after the creation of the pair of holes, we first expand one of the two holes vertically down and then horizontally right, as shown in Fig. 5. Following the spirit of Sec. III, consider a circuit \mathcal{G} composed of three gates of the form $g_l = \exp(i\frac{\pi}{4}Q_l)$, where Q_l are defined as:

$$\begin{aligned} Q_1 &= \sigma_{y_1} \sigma_{z_2} \sigma_{z_5} \sigma_{z_6}, \\ Q_2 &= \sigma_{y_2} \sigma_{z_3} \sigma_{z_7} \sigma_{z_8}, \\ Q_3 &= \sigma_{y_4} \sigma_{z_3} \sigma_{z_9} \sigma_{z_{10}}. \end{aligned} \quad (36)$$

In this case, the state is stabilized by $Z_{s_2}, Z_{s_3}, Z_{s_4}$ (and other stabilizer generators) with logical operators X_L and Z_{s_1} . The

$\mathbb{L}_1(t_0)$	Z_{s_1}	$\xRightarrow{g_1}$	$\mathbb{L}_1(t_1)$	$Z_{s_1}Z_{s_2}$	$\xRightarrow{g_2}$	
$\mathbb{L}_2(t_0)$	X_L		$\mathbb{L}_2(t_1)$	X_L		
$S_1(t_0)$	Z_{s_2}		$S_1(t_1)$	σ_{x_1}		
$S_2(t_0)$	Z_{s_3}		$S_2(t_1)$	Z_{s_3}		
$S_3(t_0)$	Z_{s_4}		$S_3(t_1)$	Z_{s_4}		

$\mathbb{L}_1(t_2)$	$Z_{s_1}Z_{s_2}Z_{s_3}$	$\xRightarrow{g_3}$	$\mathbb{L}_1(t_3)$	$Z_{s_1}Z_{s_2}Z_{s_3}Z_{s_4}$		
$\mathbb{L}_2(t_2)$	X_L		$\mathbb{L}_2(t_3)$	X_L		
$S_1(t_2)$	σ_{x_1}		$S_1(t_3)$	σ_{x_1}		
$S_2(t_2)$	σ_{x_2}		$S_2(t_3)$	σ_{x_2}		
$S_3(t_2)$	Z_{s_4}		$S_3(t_3)$	σ_{x_3}		

TABLE I. The related transformation of stabilizer generators $\{S_i\}$ and logical operators $\{\mathbb{L}_i\}$ of a Z -cut qubit in Fig. 5 is shown under gate operation $\{g_i\}$.

transformation of the stabilizer generators and logical operators under \mathcal{G} is listed in Table. I. We can see that the circuit \mathcal{G} maps logical operator X_L and Z_L to X_L'' and Z_L'' in panel (c) of Fig. 5, and also maps the system Hamiltonian in panel (a) to the ones shown in panel (c).

Now we transform this procedure to an adiabatic one that gives the same state evolution following Theorem 1. Set $U_l(t, t_{l-1}) = \exp(i\pi/4 f_l(t) Q_l)$ for time segment $t \in [t_{l-1}, t_l]$, and adiabatically deform the Hamiltonian as in Eq. (28). Note that Q_1 only anticommutes with the Z_{p_2} term in the system Hamiltonian, which guarantees that even if errors occur, the adiabatic evolution is still valid (Lemma 2). The situation is the same for Q_2 and Q_3 . We first consider the adiabatic transformation generated by $U_1(t, t_0)$:

$$H(t) = -J \cos[f_1(t)] Z_{p_2} - J \sin[f_1(t)] \sigma_{x_1} - J \sum_{j \neq 1, 2} Z_{p_j} - J \sum_i X_{s_i}, \quad (37)$$

for $t \in [t_0, t_1]$, with

$$H(t_1) = -J \sigma_{x_1} - J \sum_{j \neq 1, 2} Z_{p_j} - J \sum_i X_{s_i}. \quad (38)$$

At this time, qubit 1 is in the state $|+\rangle$. For U_2 and U_3 , we see that Q_2 commutes with Q_3 , while Q_2 only anticommutes with Z_{p_3} , and Q_3 only anticommutes with Z_{p_4} . According to Lemma 3, the adiabatic procedures generated by U_2 and U_3 can be done simultaneously with the same state transformation as if done serially. The corresponding Hamiltonian deformation is

$$H(t) = -J \sigma_{x_1} - J \cos[f_2(t)] Z_{p_3} - J \sin[f_2(t)] \sigma_{x_2} - J \cos[f_2(t)] Z_{p_4} - J \sin[f_2(t)] \sigma_{x_4} - J \sum_{j \neq 1, 2, 3, 4} Z_{p_j} - J \sum_i X_{s_i} \quad (39)$$

for $t \in [t_1, t_2]$, with

$$H(t_2) = -J \sigma_{x_1} - J \sigma_{x_2} - J \sigma_{x_4} - J \sum_{j \neq 1, 2, 3, 4} Z_{p_j} - J \sum_i X_{s_i}, \quad (40)$$

with qubits 1, 2, 3, and 4 all in the state $|+\rangle$, while they are all protected from σ_z errors by the energy gap.

This procedure can be generalized to obtain arbitrarily large square hole with distance equal to the perimeter d (assuming d is a multiple of 4). We first adiabatically expand $d/4$ times vertically down to form a long strip like that in panel (b) of Fig. 5, and then adiabatically expanding horizontally right parallel $d/4$ times as in panel (c). In all, we need about $d/2$ time steps of adiabatic evolution.

2. Error propagation

Even though the ground space is protected by an energy gap, there is still a nonzero probability that thermal excitations will occur at finite temperature. In this section, we apply the result of Lemma 2 to study the propagation of these errors. If errors occur outside the hole or inside the hole, they will not be affected by the adiabatic process at all. However, if errors occur on the boundary of the hole before the adiabatic process, they may potentially propagate during the adiabatic procedure and cause uncorrectable logical errors. Consider the case in Fig. 6. Before expanding the hole verti-

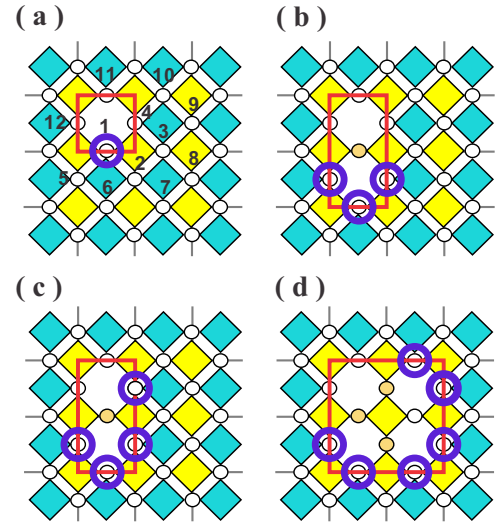


FIG. 6. (Color online) Error propagation during an adiabatic process to enlarge a hole of an X -cut logical qubit. Colored squares indicate that the corresponding X_s , σ_x (yellow) and Z_p , σ_z (cyan) operators are turned on. The purple circle around a qubit indicates a σ_z error occurs on that qubit. (a) A σ_z error occurs on qubit 1. (b) Effective errors after the adiabatic process. (c) An additional σ_z error occurs on qubit 4. (d) Effective errors after the adiabatic procedure to enlarge the hole will cause a logical error after decoding.

cally down, assume a σ_z error occurs on qubit 1. Then according to Lemma 2, the σ_z error will propagate to $\sigma_{x_1}\sigma_{z_2}\sigma_{z_5}\sigma_{z_6}$, as shown in panel (b). The effective errors are $\sigma_{z_2}\sigma_{z_5}\sigma_{z_6}$, since σ_{x_1} has no effect because state of qubit 1 is $|+\rangle$. However, if another σ_z error occurs on qubit 4, as shown in panel (c), then after expanding horizontally rightward, we get effective errors $\sigma_{z_5}\sigma_{z_6}\sigma_{z_7}\sigma_{z_8}\sigma_{z_9}\sigma_{z_{10}}$, which occupy majority of the qubits around the hole. If the minimum-weight error correction is taken, it will close the path by applying $\sigma_{z_{11}}\sigma_{z_{12}}$ and cause a logical Z error. So in general, this procedure is not fault-tolerant by the meaning of Def. 1. However, we can get around this problem by the following observation: if before the hole expansion, the system is prepared in the $|0_{DL}^Z\rangle$, then a logical Z error has no effect on the state. The situation is the same for the $|+\rangle_{DL}^X$ state for an X -cut double qubit. Fortunately, as we will see later, in this scheme we only need to expand a Z -cut hole after creation a $|0_{DL}^Z\rangle$ state and X -cut hole after creation a $|+\rangle_{DL}^X$ state, so the non fault-tolerance of this procedure can be overcome.

C. Moving logical qubits

We now turn to the realization of logical gate operations in surface codes, like logical CNOT, S , Hadamard and T gates. An element way to do these logical gates is by adiabatically moving the holes around each other on a single 2D lattice. In this section, we focus on the details of hole movement by adiabatically deforming the system Hamiltonian. We start with a scheme free of errors at first and then discuss the corresponding error propagation and fault-tolerance.

1. Scheme

We focus on the Z -cut qubit in this section, the method for the X -cut is similar. Consider a Z -cut qubit hole as shown in Fig. 7. Initially, the system Hamiltonian is

$$H(t_0) = -J \sum_{i=5}^8 \sigma_{x_i} - J \sum_{i=12}^{14} \sigma_{x_i} - J \sum_{j=1}^4 Z_{p_j} + H_{\text{rest}}, \quad (41)$$

where H_{rest} represents terms which are not altered in this process but are shown in Fig. 7. We start with a circuit \mathcal{G} composed of gates $\{g_l\}$ generated by $\{Q_l\}$. For illustration purposes, we divide them into two groups. We first expand the hole horizontally right as shown from panel (a) to panel (b), and then we shrink the hole rightward, as shown from panel (c) to panel (d). Consider the expansion procedure generated by:

$$Q_l = i\sigma_{x_l} Z_{p_l}, \quad 1 \leq l \leq 4, \quad (42)$$

and the corresponding unitary transformations of the Hamiltonian $U_l = \exp(i f_l(t) Q_l)$, for l from 1 to 4. We can see

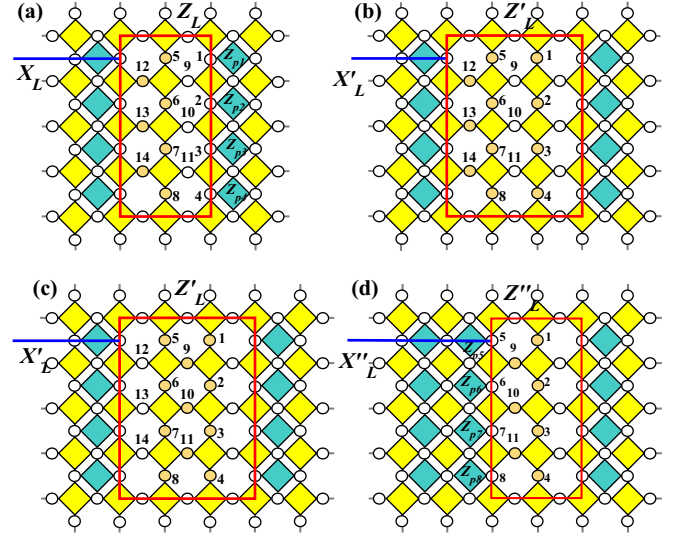


FIG. 7. (Color online) Adiabatic process for moving a Z -cut logical qubit hole horizontally right. Colored squares indicate that the corresponding X_s , σ_x (yellow) and Z_p , σ_z (cyan) operator are turned on. Logical operators of the qubit are X_L and Z_L in (a). An adiabatic process between (a) and (b) maps X_L to X'_L and Z_L to Z'_L . Similarly, an adiabatic process between (c) and (d) maps X'_L to X''_L and Z'_L to Z''_L .

that each Q_l anticommutes only with Z_{p_l} , so we can apply the adiabatic procedures generated by Q_1, Q_2, Q_3, Q_4 simultaneously

$$H(t) = -J \sum_{j=1}^4 \{ \cos[f_1(t)] Z_{p_j} + \sin[f_1(t)] \sigma_{x_j} \} - J \sum_{i=5}^8 \sigma_{x_i} - J \sum_{i=12}^{14} \sigma_{x_i} + H_{\text{rest}}, \quad (43)$$

for $t \in [t_0, t_1]$, and obtain

$$H(t_1) = -J \sum_{i=1}^4 \sigma_{x_i} - J \sum_{i=5}^8 \sigma_{x_i} - J \sum_{i=12}^{14} \sigma_{x_i} + H_{\text{rest}} \quad (44)$$

at time t_1 as shown in panel (b). At this time, all qubits inside the hole are set to the $|+\rangle$ state. To contract the hole, rightward, we follow the circuit generated by Q_l ,

$$Q_l = iZ_{p_l} \sigma_{x_l}, \quad 5 \leq l \leq 8, \quad (45)$$

and the corresponding unitary transformation of the Hamiltonian $U_l = \exp(i f_l(t) Q_l)$. We need to be a little careful here, since Q_l here anticommutes with two terms in the Hamiltonian. For example, $i\sigma_{x_5} Z_{p_5}$ anticommutes with both σ_{x_5} and $\sigma_{x_{12}}$. To get around this, we turn off the terms $-J\sigma_{x_{12}}$, $-J\sigma_{x_{13}}$, $-J\sigma_{x_{14}}$ in the above equation, and turn on $-J\sigma_{x_9}$, $-J\sigma_{x_{10}}$, $-J\sigma_{x_{11}}$ instead. We can see that this procedure doesn't change the state of the system and can be done either adiabatically or instantaneously, making the Hamiltonian

to be:

$$H'(t_1) = -J \sum_{i=1}^{11} \sigma_{x_i} + H_{\text{rest}}. \quad (46)$$

Q_l now anticommutes with just one stabilizer generator (which is σ_{x_l}). Like the expansion process, we can adiabatically deform the Hamiltonian:

$$H(t) = -J \sum_{j=5}^8 \{ \cos[f_2(t)] \sigma_{x_j} + \sin[f_2(t)] Z_{p_j} \} \\ - J \sum_{i=1}^4 \sigma_{x_i} - J \sum_{i=9}^{11} \sigma_{x_i} + H_{\text{rest}}, \quad (47)$$

for $t \in [t_1, t_2]$, and obtain

$$H(t) = -J \sum_{j=5}^8 Z_{p_j} - J \sum_{i=1}^4 \sigma_{x_i} - J \sum_{i=9}^{11} \sigma_{x_i} + H_{\text{rest}}, \quad (48)$$

which completes a full cycle of hole movement and leaves us ready for the next cycle of Hamiltonian deformation. The original ground space will be mapped to the one with a hole sitting one unit rightward of the original one (see panel (d)), and X_L and Z_L will be mapped to X'_L and Z'_L following Theorem. 1.

Remark 2. Note that the two steps of the adiabatic expansion and contraction of the hole can be combined into one step, if we turn off $-\sigma_{x_{12}}, -\sigma_{x_{13}}, -\sigma_{x_{14}}$ while turning on $-\sigma_{x_9}, -\sigma_{x_{10}}$ and $-\sigma_{x_{11}}$ at the beginning. So we need just one time step to adiabatically deform the system Hamiltonian to move a hole by one unit.

2. Error propagation and fault tolerance

Like the case of hole enlargement, there's chance that thermal errors will cause an excitation. Errors outside or inside the holes will not be propagated by the process. However, if errors occur on the boundary of the hole before moving, they may potentially propagate to uncorrectable logical errors. Consider the case in Fig. 8 for a 2 units movement rightward. Before expanding the hole horizontally right, assume σ_z errors occurs on qubit 1 and qubit 2, as shown in panel (a). They will be propagated to:

$$\sigma_{z_1} \sigma_{z_2} \mapsto \sigma_{z_3} \sigma_{z_4} \sigma_{z_5} \sigma_{z_6} \sigma_{z_7} \sigma_{z_8} \sigma_{z_9} \sigma_{z_{10}} \sigma_{z_{11}} \sigma_{z_{12}}, \quad (49)$$

by the subsequent adiabatic operation, as shown in panel (b). If we keep expanding the hole rightward, the errors will occupy more than half of the qubits on the perimeter of the hole, and cause a logical Z error after later decoding. Similarly, if σ_x errors occur on qubit 1 and qubit 2, the effective errors after the adiabatic procedure will be

$$\sigma_{x_1} \sigma_{x_2} \mapsto \sigma_{z_1} \sigma_{z_2} \sigma_{z_3} \sigma_{z_4} \sigma_{z_5} \sigma_{z_6} \sigma_{z_7} \sigma_{z_8} \sigma_{z_9} \sigma_{z_{10}} \sigma_{z_{11}} \sigma_{z_{12}}, \quad (50)$$

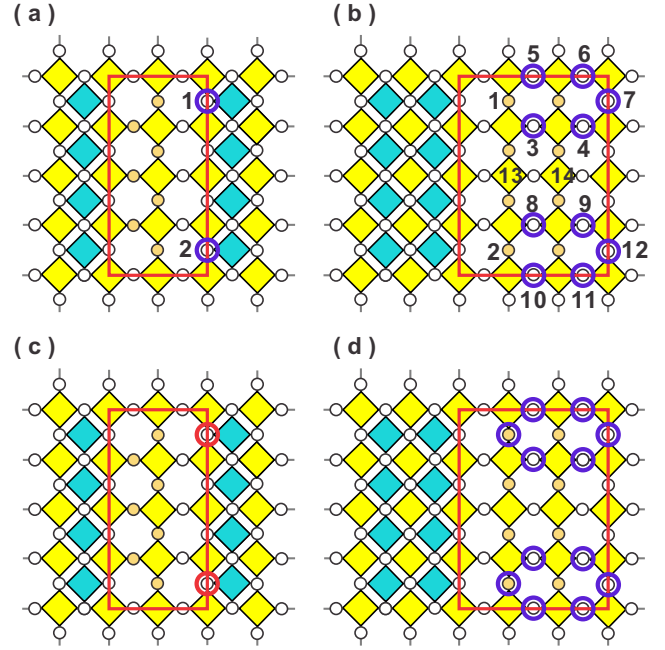


FIG. 8. (Color online) Error propagation during adiabatic process to move a hole of an X -cut logical double qubit horizontally right. Colored squares and qubits indicate that the corresponding X_s , σ_x (yellow) and Z_p , σ_z (cyan) operators are turned on. The purple circle around qubit indicates a σ_z error occurs on that qubit and a red one indicates a σ_x error occurs. (a) σ_z errors occur on qubit 1 and 2. (b) Effective errors caused by σ_{z_1} and σ_{z_2} after adiabatic process. (c) σ_x errors occur on qubit 1 and 2. (d) Effective errors caused by σ_{x_1} and σ_{x_2} after adiabatic process.

as shown in panel (d). In general, the adiabatic procedure to move the hole on its own is not fault-tolerant, since the circuit \mathcal{G} we follow to build the adiabatic procedure is not a fault-tolerant one, and the results from Ref. [24] cannot be used here directly.

Fortunately, we can still make this process fault-tolerant. Errors that occur on the boundary of the hole, like qubit 1 and qubit 2 in this example, can be detected after each step of hole movement by measuring the qubits inside the hole after the expansion, since they are correlated, as shown in Fig. 8. In this case, we will do σ_x measurement on qubit 3, 4, 8, 9, 13 and 14, when we are in panel (b). If any of these measurements give -1 , it indicates that errors (which could be σ_x or σ_z) occurred on the boundary's right side before the hole expansion, and we need to turn off the system Hamiltonian and do a full cycle of syndrome measurement and error correction before they become uncorrectable. A σ_z error happens on the boundary with probability about $\exp(-4c\beta J)$ per time step, while σ_x happens on the boundary with probability about $\exp(-2c\beta J)$, so the probability that we must do a full cycle of error correction during hole movement is low.

In practice, measurements themselves involve errors whose effect was discussed earlier in this section. Here, we need to

check the probability that the measurement outcomes cause us to make a wrong decision about error correction. As an example, if a σ_z error occurs on qubit 1 in panel (a), qubit 3 and 4 in panel (b) will not be protected by an energy gap, and we assume that the probability of a wrong measurement outcome in these cases is p each time step. Meanwhile, if a σ_x error occurs on qubit 1 in panel (c), qubit 3 and 4 in panel (d) are protected by an energy gap $4J$. Fortunately, we can make the

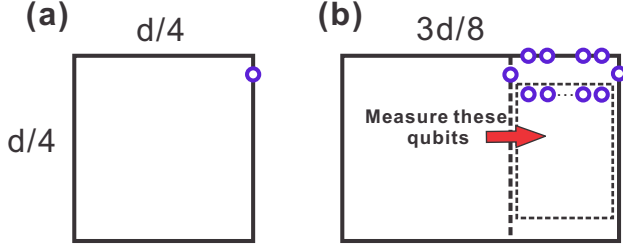


FIG. 9. (Color online) Scheme to fault-tolerantly detect errors occurring on the boundary. (a) Before the movement, an error occurs on the boundary. (b) After expanding the hole $d/8$ units rightward, the error propagates to a strip of errors. We measure all qubits in the dashed box and determine if the corresponding error happened on the boundary based on the majority vote of the measurement outcomes of each row of qubits.

uncorrectable error rate arbitrarily small by growing the lattice size and hole size, using majority vote. Consider a square hole with perimeter d as shown in panel (a) of Fig. 9. Now we expand the hole $d/8$ units rightward and measure σ_x in the dashed area of panel (b). The number $d/8$ is chosen so that error detection can be applied before an error can propagate to an uncorrectable error. If an error occurs on the boundary of the hole before moving, it will corrupt an entire row of qubits in the dashed area of panel (b) in Fig. 9. So, for each row of qubits, we do a majority vote based on the measurement outcomes to determine whether an error happened on the boundary. For any row, if more than half of the measurement outcomes are -1 , we infer that an corresponding error occurred at the boundary of the hole before moving, and therefore error correction must be applied. Let E be the event that errors happened on the boundary before movement, and let D be the event that we decide to do decoding and error correction based on the majority vote. Then the probability that such errors occurred on the boundary and is not detected is roughly

$$P_L = P(\bar{D}, E) = P(\bar{D}|E)P(E) \sim O\left(\frac{d}{4}p^{\lfloor \frac{d}{16} \rfloor + 1}e^{-4c\beta J}\right). \quad (51)$$

Here $d/4$ indicates that misidentification can occur on any of $d/4$ rows. This gives a rough bound on the probability of logical errors during the $d/8$ unit hole movement.

On the other hand, the probability that no error occurred on the boundary, but we do an unnecessary decoding can be

estimated as

$$P_U = P(D, \bar{E}) = P(D|\bar{E})P(\bar{E}) \sim O\left(\frac{d}{4}p^{\lfloor \frac{d}{16} \rfloor}e^{-4c\beta J}\right). \quad (52)$$

We can see that both P_L and P_U can be made arbitrarily small with the growth of hole size, and thus the adiabatic movement process can be rendered fault-tolerant.

Remark 3. We only analyzed the error propagation for the case of hole expansion. It is worth noting that for the procedure to adiabatically contract the hole, errors occurring on the boundary of the hole will *not* accumulate to uncorrectable logical errors, and thus can be left for future error correction.

D. Creation of $|0\rangle$ ($|+\rangle$) state for X (Z)-cut double qubit

The second type of logical state initialization is to prepare the $|0\rangle$ state for an X -cut qubit or $|+\rangle$ state for a Z -cut qubit. We show an example for an X -cut qubit in detail. For a Z -cut qubit, the procedure is similar.

This can be done using a logical Hadamard after initializing the $|+\rangle$ state for X -cut qubit. However, we have not shown how to perform a logical Hadamard yet, and it is also extremely useful to directly initialize the $|0\rangle$ state for an X -cut qubit, as we will see in next few sections.

Suppose we have created a $|+\rangle$ state for an X -cut qubit with two holes attached to each other, as shown in panel (a) of Fig. 10. The logical Z operator in this case can be $\sigma_{z_1}, \sigma_{z_2}$

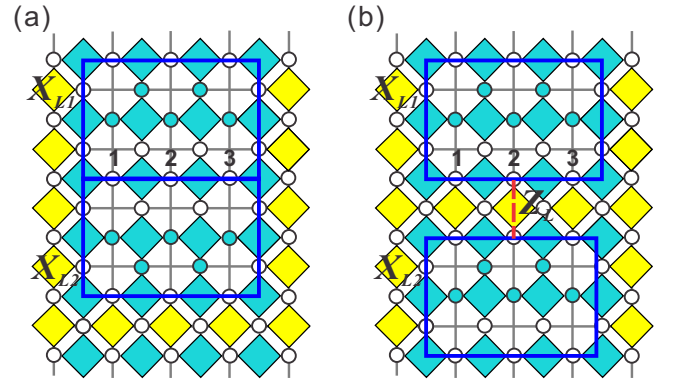


FIG. 10. (Color online) Creation of $|0\rangle$ state for X -cut double qubit. (a) Create a $|+\rangle_{DL}^X$ with two holes attached to each other. Measure $\sigma_{z_1}, \sigma_{z_2}$ and σ_{z_3} and do majority vote to determine whether $|0\rangle_{DL}^X$ or $|1\rangle_{DL}^X$ is prepared. (b) Move two holes apart to increase error correction ability of σ_z errors of the logical qubit. Note that both σ_z and σ_x errors on qubits between two holes during the adiabatic movement have no uncorrectable effect on logical state $|0\rangle_{DL}^X$ or $|1\rangle_{DL}^X$ and can be left for future error correction.

or σ_{z_3} , and they all commute with the system Hamiltonian. If we measure any one of them, we can prepare the logical state $|0\rangle$ or $|1\rangle$. Either one is useful as long as we know which state

it is for certain. If any σ_z errors occur on these qubits, it will have no effect, and any single σ_x errors on these qubits suffers an energy penalty of $4J$ and leaves Z_p operators nearby flipped and correctable by a future error correction procedure. However, when a σ_x happens on these qubits, it will give an incorrect measurement outcome, and will affect any future operations conditioned on whether the state is $|0\rangle$ or $|1\rangle$. This can also be resolved by measuring σ_z on all three qubits and taking the majority vote to determine the measurement outcome. This procedure can be extended to the square hole with perimeter d , where there are $d/4$ qubits shared by two holes. Note that the first measurement error is suppressed by the energy penalty, and occurs with probability $\exp(-4c\beta J)$, while the subsequent measurement errors may not suffer an energy penalty. We assume that the probability to obtain a wrong measurement result is p each time step. The probability that we prepare a $|0_{DL}^X\rangle$ ($|1_{DL}^X\rangle$) state with an erroneous measurement -1 ($+1$) can be estimated to be:

$$P_L \sim O(p^{\lfloor \frac{d}{8} \rfloor + 1} e^{-4c\beta J}), \quad (53)$$

which decreases rapidly with the growth of the hole size, and can be made arbitrarily small. After the measurement, we separate the two holes by distance d , as illustrated in panel (b) of Fig. 10 for a single time step of movement. It takes about $d/2$ time steps in total to move the pairs of holes apart by distance d if the two holes move simultaneously. Any σ_x and σ_z errors on qubit 1, 2, 3 will not propagate to uncorrectable errors during the movement. The hole movement process can be done adiabatically and fault-tolerantly with gap protection, as described in the previous section. Thus, the whole state preparation process can be made fault-tolerant.

E. Logical Z (X) measurement for X (Z)-cut double qubit

Like the case of initialization, there are two types of measurement procedures. The first is measuring in the Z (X) basis for an X (Z)-cut qubit while the second is measuring in the Z (X) basis for a Z (X)-cut qubit.

The first type of measurement is essentially the reverse process of creating the state $|0\rangle$ ($|+\rangle$) for an X (Z)-cut qubit. For an X -cut qubit shown in Fig. 10, we first move two holes that are initially d units apart together to contact each other, and then measure σ_z on all qubits shared by the two holes and take a majority vote of the outcomes. After that, we separate the two holes back to their original positions. Note that unlike traditional measurement-based QC on the surface code, this measurement is non-destructive and we do not annihilate the holes. The measurement procedure can also be viewed as a logical state preparation that will be used in the future computation. The second type of measurement procedure will be discussed in Sec. IV G.

F. Holonomic Logical CNOT

The logical CNOT gate is one of the most important logical operations in the surface code HQC scheme. Based on our results on adiabatic hole movement, we can realize the logical CNOT gate. In this section, we show that by adiabatically braiding one hole around a different type of hole, we can get a closed loop holonomy which can be recognized as a logical CNOT. Starting from panel (a) of Figs. 11 and 12, the adiabatic movement procedure is shown in details from panel (b) to panel (f). In Fig. 11, following the discussion in Sec. IV C and Theorem. 1, $X_{L_1} \otimes I_{L_2}$ transforms to $X_{L_1} \otimes X_{L_2}$ up to a multiplication by X_s stabilizer generators inside the dashed square. We can conclude that X_L operators transform in the following way:

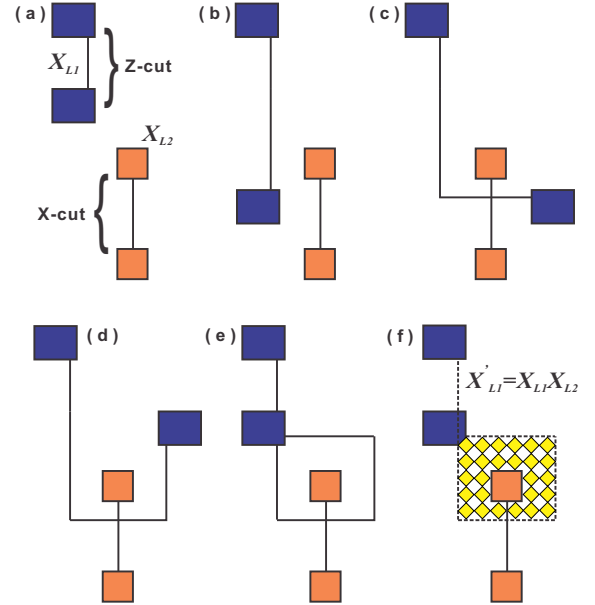


FIG. 11. (Color online) Adiabatic braiding process of a Z -cut hole (dark blue) around an X -cut hole (orange). The operator X_{L_1} has been stretched to multiply a loop of σ_x operators which is equivalent to X_{L_2} up to multiplication by X_s stabilizer generators (yellow) inside the loop, while X_{L_2} remains the same under the transformation.

$$\begin{aligned} X_{L_1} \otimes I_{L_2} &\rightarrow X_{L_1} \otimes X_{L_2}, \\ I_{L_1} \otimes X_{L_2} &\rightarrow I_{L_1} \otimes X_{L_2}. \end{aligned} \quad (54)$$

Similarly, from Fig. 12, we can see that $I_{L_1} \otimes Z_{L_2}$ transforms to $Z_{L_1} \otimes Z_{L_2}$ up to multiplication by Z_p stabilizer generators inside the strip. The Z_L operators transform as:

$$\begin{aligned} Z_{L_1} \otimes I_{L_2} &\rightarrow Z_{L_1} \otimes I_{L_2}, \\ I_{L_1} \otimes Z_{L_2} &\rightarrow Z_{L_1} \otimes Z_{L_2}. \end{aligned} \quad (55)$$

The closed loop adiabatic evolution can be recognized as a closed loop holonomy which gives a logical CNOT with a

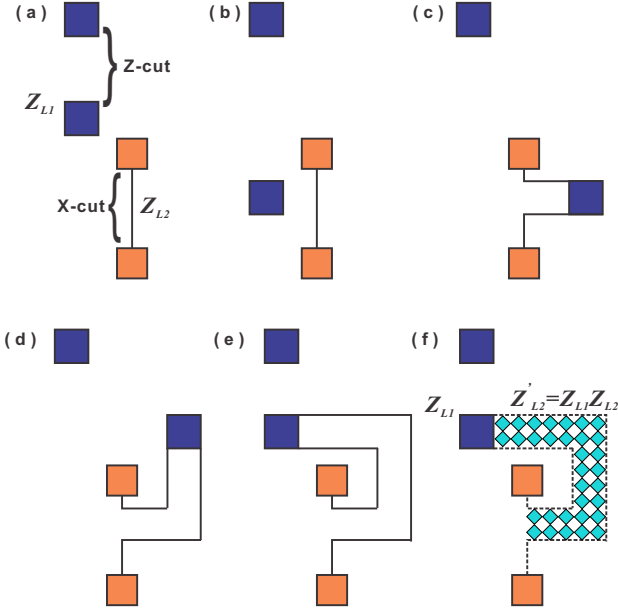
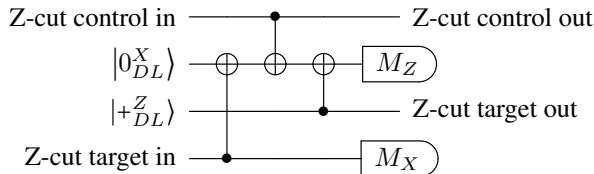


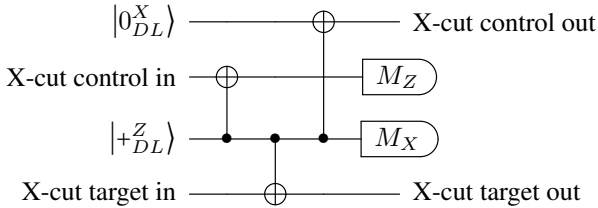
FIG. 12. (Color online) Adiabatic braiding process of a Z -cut hole (dark blue) around an X -cut hole (orange). The operator Z_{L2} has been stretched to form a strip of σ_z operators, which is equivalent to Z_{L1} up to multiplication by Z_p stabilizer generators (cyan) inside the strip, while Z_{L1} remains the same under the transformation.

Z -cut qubit as the control and an X -qubit as the target. It also reflects the topological property of braiding on 2D lattice since local deformation of movement path does not have effects on the state. Note that the fault-tolerance of this operation is guaranteed by the fault-tolerance of adiabatic hole movement.

CNOTs from Z -cut qubits to X -cut qubits are not enough. We need to extend to CNOTs between logical qubits of the same type. For Z -cut qubits, we have the following circuit:

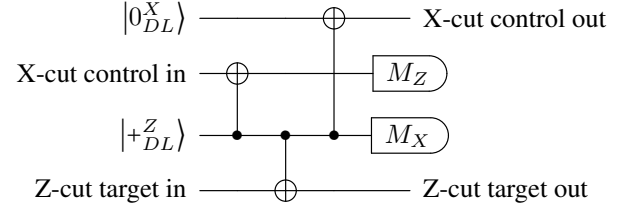


which is equivalent to $Z_L^{(1-M_X)/2}$ on the target qubit followed by a CNOT, then followed by $X_L^{(1-M_Z)/2}$ on the target qubit. Similarly, the CNOT between two X -cut logical qubits can be built from following circuit:



up to a correction of logical X s and Z s. The last kind of CNOT, with an X -cut qubit as control and a Z -cut as target,

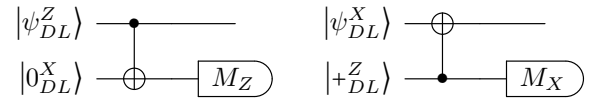
can be obtained from the circuit realizing CNOT between Z -cut qubits:



Note that for all four different logical CNOTs, the building block is the CNOT from Z -cut to X -cut. In addition, we also need to prepare ancillas in logical $|0_{DL}^X\rangle$ and $|+_{DL}^Z\rangle$ (which is shown in Sec. IV D), and to do Z measurements of X -cut qubits and X measurements of Z -cut qubit (as discussed in Sec. IV E). All of these procedures can be done fault-tolerantly, and thus make all kinds of logical CNOT fault-tolerant.

G. Measurement of Z (X) basis for Z (X)-cut double qubit

This type of measurement is necessary when doing state distillation (discussed later). Naively, this process can be done by contracting the size of the hole and doing stabilizer measurements. However, stabilizer measurement is not compatible with the system Hamiltonian. What is worse, we close the hole after the measurement to destroy the logical qubit, and we cannot reuse it later. To avoid these problems, we can use the following circuits for Z and X measurement of Z -cut and X -cut qubits, respectively:



These circuits take an ancilla state $|0_{DL}^X\rangle$ or $|+_{DL}^Z\rangle$, and a logical CNOT with a Z -cut qubit as the control and an X -cut qubit as the target, which can both be realized fault-tolerantly. Thus, this type of measurement procedure is fault-tolerant. Note that, like the measurement of the first type in Sec. IV E, this measurement procedure doesn't annihilate the hole after measurement. The ancilla qubits after measurement are effectively prepared to $|0_{DL}^X\rangle$ (or $|+_{DL}^Z\rangle$) fault-tolerantly, which can be used again as ancillas for future computation.

H. Ancilla recycling

As we have seen so far, to implement different types of CNOTs, we need to frequently create and measure logical qubits. Moreover, state distillation procedures also need large number of fresh ancilla qubits and logical state measurements. We have discussed two different types of state creation— $|0\rangle$ ($|+\rangle$) for X (Z)-cut and $|0\rangle$ ($|+\rangle$) for Z (X)-cut—and two different types of measurement— X (Z) measurement for Z (X)-cut qubit and Z (X) measurement for Z (X)-cut qubit. All can be done fault-tolerantly with constant gap protection, and both kinds of logical state measurement can

be made non-destructive, so states after measurement can be reused as ancillas to avoid having to create a new logical qubits. This is particularly important, as we have seen that to create a logical state we need to turn off some X_s or Z_p operators, whose eigenvalues are uncertain when stabilizer Hamiltonian is turned on. With this ancilla recycling process, we can prepare all logical qubits, data or ancilla, right after we turn on the system Hamiltonian at the very beginning of the computation and never create new logical qubits during the computation.

I. State injection

As will be seen in Sec. IV K and IV L, to get the logical S , T and Hadamard gates, we need to create particular logical ancilla states $|Y_{DL}\rangle = \frac{1}{\sqrt{2}}(|0_{DL}\rangle + i|1_{DL}\rangle)$ and $|A_{DL}\rangle = \frac{1}{\sqrt{2}}(|0_{DL}\rangle + e^{i\pi/4}|1_{DL}\rangle)$. However, there's no obvious way to perform arbitrary rotation of logical qubit with large distance and local Hamiltonians transformation. To deal with this problem, we need to create a logical qubit in which the logical Z operator is just one σ_z on single qubit, with the stabilizer Hamiltonian turned on. We focus on X -cut dou-

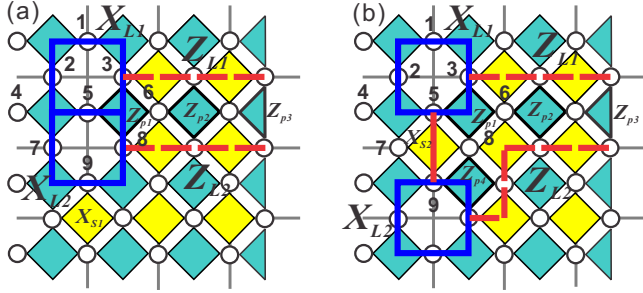


FIG. 13. (Color online) State injection for a X -cut qubit. Colored squares indicate that the corresponding X_s (yellow) and Z_p (cyan) operator is turned on.

ble qubits. We first put an *existing* X -cut qubit into the state $|+\rangle_{DL}^X$ with the two holes attached to each other, as in panel (a) of Fig. 13. This can be done by doing a logical X measurement on an existing X -cut qubit (Sec. IV G) and moving the two holes together. Without loss of generality, assume the state after measurement to be $|+\rangle_{DL}^X$. Note that $Z_L = \sigma_{z_5}$ is equivalent to $Z_{L_1}Z_{L_2}$ up to multiplication by Z_p operators, as shown in panel (a), which gives:

$$\sigma_{z_5} = Z_{p_1}Z_{p_2}Z_{p_3}Z_{L_1}Z_{L_2}. \quad (56)$$

For the $|\pm\rangle_{DL}^X$ state, the effect of σ_{z_5} is

$$\begin{aligned} \sigma_{z_5}|\pm\rangle_{DL}^X &= \sigma_{z_5}|\pm\rangle_{SL}^X|1\rangle_{SL}^X|2\rangle_{SL}^X \\ &= Z_{L_1}Z_{L_2}|\pm\rangle_{SL}^X|1\rangle_{SL}^X|2\rangle_{SL}^X \\ &= |\mp\rangle_{SL}^X|1\rangle_{SL}^X|2\rangle_{SL}^X. \end{aligned} \quad (57)$$

Applying a pulse $V_c = g\sigma_{z_5}$ for a short time τ , with Hamiltonian

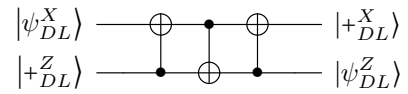
$$H = H_{\text{stab}} + V_c, \quad (58)$$

we can see that $[V_c, H_{\text{stab}}] = 0$, where H_{stab} is the stabilizer Hamiltonian shown in panel (a). The pulse will not cause a transition from the ground space to another eigenspace of H_{stab} . If τ is chosen such that $g\tau = \theta/2$, we have the state evolution:

$$\begin{aligned} &\exp\left(-i\frac{\theta}{2}\sigma_{z_5}\right)|+\rangle_{SL}^X|1\rangle_{SL}^X|2\rangle_{SL}^X \\ &= \frac{e^{-i\frac{\theta}{2}}}{\sqrt{2}}\left(\frac{|+\rangle_{SL}^X|1\rangle_{SL}^X|2\rangle_{SL}^X + |-\rangle_{SL}^X|1\rangle_{SL}^X|2\rangle_{SL}^X}{\sqrt{2}}\right. \\ &\quad \left.+ e^{i\frac{\theta}{2}}\frac{|+\rangle_{SL}^X|1\rangle_{SL}^X|2\rangle_{SL}^X - |-\rangle_{SL}^X|1\rangle_{SL}^X|2\rangle_{SL}^X}{\sqrt{2}}\right) \\ &= \frac{e^{-i\frac{\theta}{2}}}{\sqrt{2}}(|0_{DL}^X\rangle + e^{i\theta}|1_{DL}^X\rangle), \end{aligned} \quad (59)$$

which gives the desired state we want to inject. Note that if a σ_{x_5} error occurs, it will suffer from the energy penalty, and cause the Z_p s adjacent to it to be flipped, leaving the syndrome for future error correction. On the other hand, the imprecise control of the pulse V_c can affect the state injected and cannot be detected. However, as long as rate of σ_{z_5} error is lower than a threshold, logical states $|Y_{DL}\rangle$ and $|A_{DL}\rangle$ can be obtained with sufficient precision by state distillation [48]. Then two holes can be adiabatically separated to distance d to better protect against errors, as illustrated in panel (b) of Fig. 13.

The process of state injection for a Z -cut qubit is slightly more complicated. We first inject state the $|\psi\rangle = |Y\rangle$ or $|A\rangle$ for an X -cut qubit and prepare a Z -cut qubit in state $|+\rangle$ and then we swap the state of these two logical qubits using following circuit:



Note that the $|+\rangle_{DL}^X$ is ready to be reused for state injection, and all process included here can be done fault-tolerantly.

J. State Distillation

The logical ancilla states, $|Y\rangle = |0\rangle + i|1\rangle$ and $|A\rangle = |0\rangle + e^{i\pi/4}|1\rangle$ after injection are not good enough in general for the purpose of fault-tolerant QC. Fortunately, they can be distilled to much higher fidelity [49]. The reversed encoding circuit for 7-qubit Steane code can be used to distill the $|Y\rangle$ state, with seven input logical states approximately equal to $|Y\rangle$ [39] as shown in Fig. 14. The output $|\psi\rangle$ will be closer to the logical $|Y\rangle$ state. Repeating this process multiple times, arbitrarily high fidelity $|Y\rangle$ states can be obtained exponentially quickly if the original fidelity of the input states is higher than some threshold [48]. A similar distillation circuit exists

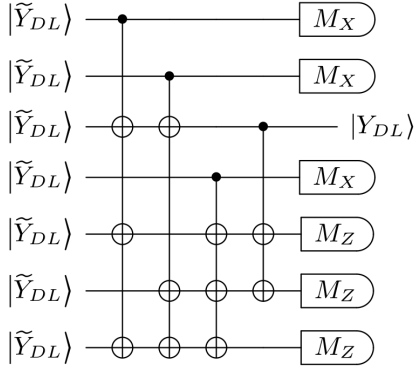


FIG. 14. Circuits for logical $|Y\rangle$ distillation from imperfect $|\tilde{Y}\rangle$ states.

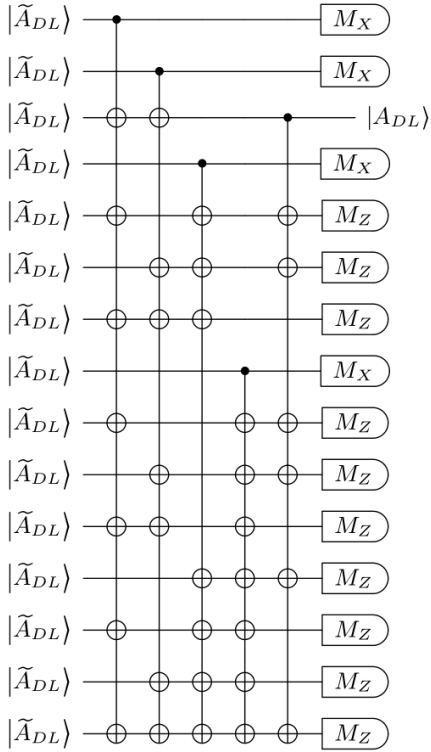
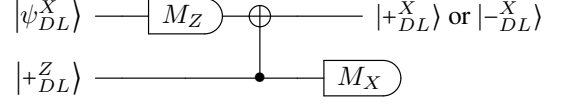


FIG. 15. Circuits for logical $|A\rangle$ distillation from imperfect $|\tilde{A}\rangle$ states.

for the $|A\rangle$ state, as shown in Fig. 15, which is the reverse of the encoding circuit for the $[[15, 1, 3]]$ truncated Reed-Muller code [39, 50]. As before, given a good enough input $|A\rangle$ state, the convergence is rapid.

Note that these distillation circuits use CNOTs between the same type of qubits, and both types of logical state measurements described in Sec. IVE and IV G. If the input states are X -cut qubits, then the logical X measurements are of the second kind, and the states after measurement are $|+_{DL}^X\rangle$ or $|-_{DL}^X\rangle$, which are ready to be reused to inject $|Y\rangle$ or $|A\rangle$ for future state distillation. The logical Z measurements are of

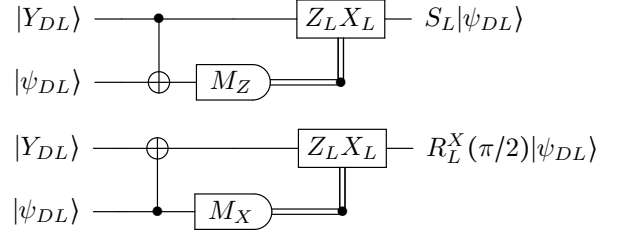
the first type, and will prepare logical states $|0_{DL}^X\rangle$ or $|1_{DL}^X\rangle$. To recycle these logical qubits to inject new $|Y\rangle$ or $|A\rangle$, we need to reset them to $|+_{DL}^X\rangle$ or $|-_{DL}^X\rangle$, which can be done by a subsequent logical X measurement:



Note that the ancilla states $|+_{DL}^Z\rangle$ or $|-_{DL}^Z\rangle$ introduced here after logical X -measurement can also be reused directly as ancilla for another logical X -measurement. The recycling process for Z -cut qubit inputs is similar.

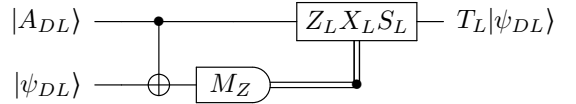
K. Logical Phase and T Gates

Given the distilled $|Y\rangle$ state, we can implement high quality logical S gates and logical $R_L^X(\pi/2) = \exp(-i\frac{\pi}{4}X_L)$ gates using the following circuits [39]:



If the measurement outcome is $+1$, nothing needs to be done; otherwise, do a ZX gate. Note that this ZX gate can be done in “software” rather than physically.

The non-Clifford gates play a central role in quantum speedup [3], and are necessary to obtain a universal gate set. For the surface code, the logical T gate is implemented with high quality distilled logical $|A\rangle$ states using this circuit [40]:



If the logical Z measurement yields a $+1$ outcome, the output state is the desired one. If the measurement yields a -1 outcome, the output is $X_L T_L^\dagger |\psi_{DL}\rangle$ and $Z_L X_L S_L$ needs to be applied to get T_L . Again, the logical X and Z gate can be done in classical “software” rather than physically. Details of commuting X_L, Z_L through S_L and T_L for classical software control were discussed in Sec. XVI. A of Ref. [40]. As usual, the states after the measurements in these circuits can all be recycled and used as ancillas for logical CNOT gates, state injection and state distillation in future computational steps.

L. Hadamard

In the existing, measurement-based QC on the surface code, a logical Hadamard is realized by first digging a “moat” around the double logical qubits by measuring single qubits around the double hole to create a logical qubit island. On

the “island”, a logical Hadamard gate is then realized by a sequence of code deformations through single qubit and stabilizer measurements, and then the “moat” at last is repaired [40]. This version of logical Hadamard is easy and efficient enough in measurement-based QC, but difficult to implement in our system when the stabilizer Hamiltonian is turned on. Instead, the logical Hadamard gate can be done directly:

$$\text{Had} = S \cdot R^X(\pi/2) \cdot S. \quad (60)$$

Both logical S and $R^X(\pi/2)$ are fault-tolerant but heavily rely on the state distillation of logical $|Y\rangle$ state.

There is a more efficient way to do a logical Hadamard, as illustrated in Ref. [51], by introducing a nontrivial domain wall on the lattice and moving the holes across the wall. The wall can be created by shifting the geometry of the lattice along a line, as shown in Fig. 16. The five body interaction terms terminating the dislocation are called twists [51]. One can see that the insertion of two twists changes the degeneracy of ground space. This can form an additional logical qubit, which we call gauge qubit \mathcal{F} . The corresponding logical operators of this qubit are also shown in Fig. 16.

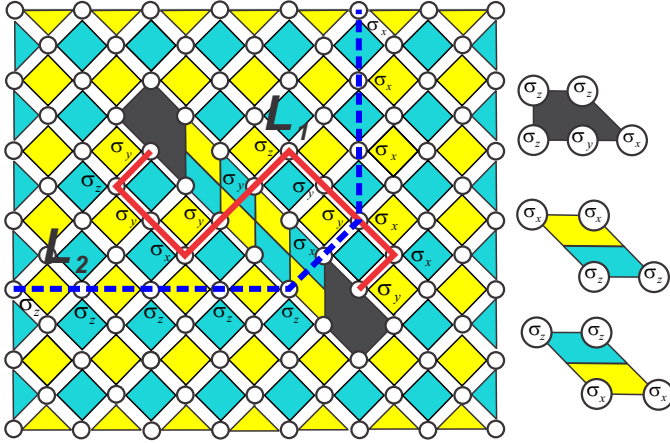


FIG. 16. (Color online) A dislocation in the geometry of the Hamiltonian produced by shifting the stabilizer generators along a line between two twists. The stabilizer generators corresponding to two different parallelograms (yellow/cyan and cyan/yellow) and a pentagon (dark gray) are shown on the right side. A pair of anticommuting strings of Pauli operators L_1 (solid red) and L_2 (dashed blue) that commute with all stabilizer generators forms the logical operators of the extra qubit \mathcal{F} attached to the pair of twists.

If a single Z (X)-cut hole is adiabatically dragged across the wall, it will change to a X (Z)-cut hole, as shown in Fig. 17. However, note that this process can also change the state of \mathcal{F} , since it will change logical operators L_1 and L_2 . This effect in general will yield additional entanglement between data qubit and \mathcal{F} . However, if we drag the second hole of the logical data qubit across the wall, it will reverse the change caused by the first hole and leave the state of \mathcal{F} unchanged. In summary, adiabatically moving two holes of a

logical qubit across the wall will give a state transformation on the data qubit:

$$|\psi_{DL}^Z\rangle \rightarrow \text{Had} |\psi_{DL}^X\rangle, \quad |\psi_{DL}^X\rangle \rightarrow \text{Had} |\psi_{DL}^Z\rangle, \quad (61)$$

for Z -cut qubits and X -cut qubits. Another problem of this

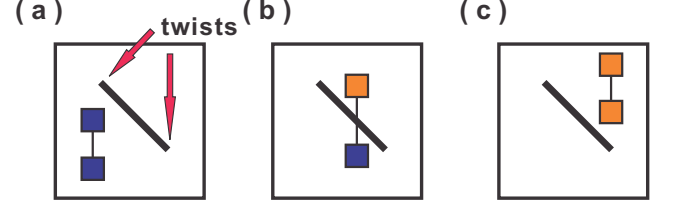
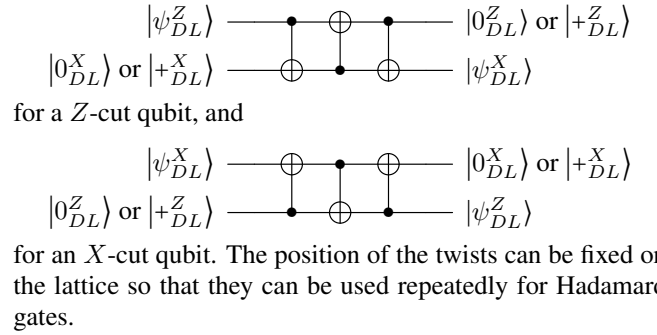


FIG. 17. (Color online) Adiabatically moving a pairs of holes of a Z -cut qubit (dark blue holes) across a twist on the surface to get a logical Hadamard gate. This process will transform a Z -cut qubit to an X -cut qubit (orange holes).

method is that it will change of the type of qubits we are working on. However, we can use an ancilla to swap the data qubit back by the circuit



V. FAULT-TOLERANCE OF THE SCHEME

We have described a way to fault-tolerantly implement QC in surface codes with a constant energy gap to suppress errors in a thermal environment. Table. II lists a summary of each procedure. Note that although adiabatic hole enlargement and state injection are not themselves fault-tolerant, they do not affect the fault-tolerance of the whole QC scheme. In addition to gap protection during the computation, fault-tolerance is guaranteed by performing single qubit and syndrome measurements before errors can propagate to become uncorrectable. We discuss the interval between syndrome measurements in Sec. V A.

So far, the error models we considered are induced by weak coupling to a thermal bath. We also need to consider other decoherence channels, which may affect qubits collectively or directly act on logical qubits. In this section, we will discuss two of them: local perturbations and adiabatic errors. In the following sections we show that they can both be exponentially bounded.

Process	Gap protection	Fault-tolerance	Dynamics	Number of time steps
Creation $ 0\rangle (+\rangle)$ for $Z (X)$ -cut qubit	Yes	Yes	Adiabatic	$\sim d/2$
Creation $ 0\rangle (+\rangle)$ for $X (Z)$ -cut qubit	Yes	Yes	Adiabatic+Measurement	$\sim d$
$Z (X)$ measurement for $X (Z)$ -cut qubit	Yes	Yes	Adiabatic+Measurement	$\sim d$
$Z (X)$ measurement for $Z (X)$ -cut qubit	Yes	Yes	Adiabatic+Measurement	$O(d)$
Hole enlargement	Yes	No	Adiabatic	$\sim d/2$
Hole movement	Yes	Yes	Adiabatic	N/A
Logical CNOT	Yes	Yes	Adiabatic+Measurement	$O(d)$
State injection	Yes	No	Adiabatic+ Pulse control	$\sim d$
State distillation	Yes	Yes	Adiabatic+Measurement	N/A
Logical S, T , Hadamard	Yes	Yes	Adiabatic+Measurement	N/A

TABLE II. Summary.

A. Error correction

A proper time period to turn off the system Hamiltonian and do error correction, in the case that there are no errors detected during the adiabatic hole movement process, is crucially important. We assume that syndrome measurement is done every m time steps, and $m \exp(-2c\beta J)$ can be regarded as the error rate on each qubit for every m time steps ($m \exp(-2c\beta J) \ll 1$), since all processes necessary for universal QC are protected by a gap of at least $2J$. Besides thermal errors accumulating on each qubit, the following types of physical errors can occur in a single syndrome measurement cycle in Sec. II A [40]:

1. σ_x error occurs when a syndrome qubit is initialized to $|0\rangle$, with probability p .
2. The Hardamard gate on syndrome qubit is not perfect. There is extra σ_x , σ_y or σ_z error following the gate, each with probability $p/3$.
3. Error occurs when a syndrome qubit is measured, with probability p .
4. CNOT gate on syndrome qubit-data qubit CNOT is not perfect, but with following erros: $I \otimes \sigma_x, I \otimes \sigma_y, I \otimes \sigma_z, \sigma_x \otimes I, \sigma_x \otimes \sigma_x, \sigma_x \otimes \sigma_y, \sigma_x \otimes \sigma_z, \sigma_y \otimes I, \sigma_y \otimes \sigma_x, \sigma_y \otimes \sigma_y, \sigma_y \otimes \sigma_z, \sigma_z \otimes I, \sigma_z \otimes \sigma_x, \sigma_z \otimes \sigma_y$ or $\sigma_z \otimes \sigma_z$, each with probability $p/15$.

Note that one needs several cycles of syndrome measurements to establish values of syndrome before actual decoding. Then, the logical error rate of surface code for m time steps with active error correction can be roughly estimated as [40]

$$P_L^m \approx d \frac{d!}{(d_e - 1)! d_e!} (m e^{-2c\beta J} + 7p)^{d_e}, \quad (62)$$

where $d_e = (d + 1)/2$. A plot of this estimate is shown in Fig. 18, for various values of $c\beta J$, p and m . We can use

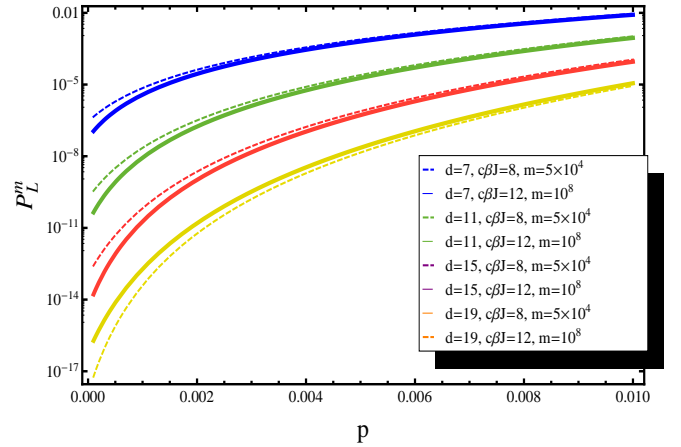


FIG. 18. (Color online) Logical error rate per m time steps for various values of m and d . The dashed lines are for $c\beta J = 8$ and solid lines for $c\beta J = 12$. The blue (top), green (second top), red (third) and yellow (bottom) lines are for $d = 7$, $d = 11$, $d = 15$ and $d = 19$, respectively.

these scaling relations to estimate the number of qubits needed to obtain a desired error rate after error correction. Our goal is that the error rate after the whole computer procedure is bounded by some particular value $\delta \ll 1$. Denoted by M the product of number of logical operation and the number of logical qubits used in an algorithm. We need to have:

$$P_L^m \lesssim \frac{m\delta}{dM}, \quad (63)$$

since each logical operation needs about d time steps in our scheme. For a particular computation like Shor's algorithm implemented on surface codes, M is of the order larger than 10^{14} [34]. We can choose $p = 0.001$ and $c\beta J = 12$, which

may be achievable in current experiments. Also, set $\delta = 0.1$, $d = 11$ and $m = 10^8$, then we have $P_L^m \approx 10^{-8}$, which satisfies the condition of Eq. (63). This requires a number of data and measurement qubits $n_{\text{tot}} = (2d-1)^2 \approx 450$ to protect a logical qubit, and perform Shor's algorithm with reasonable success probability. We can see that if large $c\beta J$ is not achievable, one can always choose a code with larger distance and more frequent error correction to compensate for the small $c\beta J$. However, if the $c\beta J$ can increase to 15, we can even reduce d to 7 and n_q to about 170, with $m = 10^{10}$ and same value of δ , making it more efficient to build a scalable QC in the near future.

B. Local perturbation

Perturbations will split the degeneracy of the ground space and cause stochastic phase errors between different logical states. This is one of the main obstacles to realizing non-Abelian holonomic quantum gates on system with a small number of qubits. However, for surface codes, the splitting of the ground space (and any other error space) caused by local perturbations will decay exponentially with the distance of the surface code, as shown by Kitaev in Ref. [10]. Actually, any system with quantum topological order is in general stable under local perturbations [31]. This might suggest that holonomic QC is more naturally suitable with systems with topological order than systems with small number of qubits. Consider a local perturbation of the general form:

$$V_{\text{local}} = - \sum_j h_j \vec{\sigma}_j - \sum_{j < p} J_{jp} (\vec{\sigma}_j \cdot \vec{\sigma}_p), \quad (64)$$

which includes all one-qubit and two-qubit interactions. The effect of V_{local} only occurs in the $d/2$ -th order of perturbation theory, and the energy splitting vanishes as

$$\Delta_{\text{split}} \sim O(Je^{-vd/2}), \quad (65)$$

where $v = \min_{ij} \{\ln(J/|h_i|), \ln(J/\|J_{ij}\|_1)\}$, which decreases quickly with growth of the code distance. Consider the case when $d = 11$, $J = 1$. To achieve an error rate of order 10^{-15} , we must to control the values of $|h_i|/J$ and $\|J_{ij}\|_1/J$ so that they are less than 10^{-3} , which is practically achievable for current or near future technology.

C. Adiabatic error

Another type of error corresponds to imperfect adiabatic evolution. We now discuss adiabatic theorem briefly and their application to bound the corresponding error. The traditional version of the adiabatic theorem stated in [47] says that the adiabatic approximation is satisfied with precision $\delta \leq \epsilon^2$ during adiabatic evolution if the condition

$$\frac{\sup_{t \in [0, T]} \|P_{s_\alpha}(t) \frac{\partial}{\partial t} H(t) P_{s_\beta}(t)\|_1}{\inf_{t \in [0, T]} K(\varepsilon_{s_\alpha}(t) - \varepsilon_{s_\beta}(t))^2} \leq \epsilon, \text{ for any } \alpha \neq \beta, \quad (66)$$

is satisfied (K is the dimension of the code space). In the case of our adiabatic process, this is equivalent to

$$\sup_{q, t \in [t_{q-1}, t_q]} \frac{\pi |\partial_t f_q(t)|}{4} \leq \epsilon \quad (67)$$

for the q th time segment. However, it is known that this statement is neither sufficient nor necessary, and we can obtain better results [52, 53]. Here we apply the result in [53] to our piecewise adiabatic evolution, serial or parallel, as described in Sec. III, for the q th time segment. We can set $T_q = t_q - t_{q-1}$, for a Hamiltonian $H(\vartheta)$ ($\vartheta = t/T_q$) that is analytic near the region $[0, 1]$ in the complex plane, with the absolute value of the imaginary part of the nearest pole being γ , and the first $\mathcal{N} \geq 1$ derivatives at boundaries equal to zero, i.e., $H^{(l)}(0) = H^{(l)}(1) = 0$ for $l \leq \mathcal{N}$. If we set

$$T_q = \frac{e}{\gamma} \mathcal{N} \frac{\xi_q^2}{\Delta_{\min}^3}, \quad (68)$$

with $\xi_q = \sup_{\vartheta \in [0, 1]} \|dH/d\vartheta\|_\infty$ (where the $\|\cdot\|_\infty$ is standard operator norm, and $\Delta_{\min} = 2J$), then the adiabatic approximation error satisfies

$$\delta_{\text{ad}} \leq (\mathcal{N} + 1)^{\gamma+1} e^{-\mathcal{N}}, \quad (69)$$

or equivalently,

$$\delta_{\text{ad}} \lesssim (c_q T_q + 1)^{\gamma+1} e^{-c_q T_q}, \quad (70)$$

with $c_q = \frac{\gamma \Delta_{\min}^3}{e \xi_q^2}$. In other words, we can decrease the adiabatic error exponentially with evolution time T_q , if it is carefully set to be proportional to \mathcal{N} and $f_q(t)$ is chosen such that a) the boundary condition mentioned above is satisfied, and b) $H(\vartheta)$ is analytic near region $[0, 1]$ on the complex plane. The adiabatic error for typical processes listed in Table. II can then be bounded by

$$\delta_{\text{ad}} \sim O\left(d \cdot \sup_q (c_q T_q + 1)^{\gamma+1} e^{-c_q T_q}\right). \quad (71)$$

So in principle, we can make adiabatic process arbitrarily small with careful chosen $\{T_q\}$ and $\{f_q\}$. Note that the thermal error rate decreases exponentially with J , while the duration of each adiabatic time segment decreases as the cube of J at fixed temperature, so the processing time overhead of an adiabatic process can be small if J is large.

Remark 4. We've analyzed that it is possible to use on the order of 10^2 physical qubits to protect a single logical qubit in practical quantum computation with protection by a constant gap enabling fault-tolerant QC in surface codes. This is quite efficient compared to the existing QC scheme in surface codes [40]. However, the assumption here is that the thermal error model is local, and the stabilizer Hamiltonian is fundamental, given by Nature. Such 4-body X_s and Z_p interactions are hard to build directly, and usually needs certain techniques, like quantum gadgets [54, 55], digital quantum simulator [56, 57], the low energy approximation from Kitaev's

honey-comb model [58] or to be generated dynamically [59]. If the Hamiltonian is effective, rather than being fundamental, it may dramatically change the local thermal error model we have assumed, and cause nonlocal errors. This possibility calls for future investigation.

VI. SUMMARY AND CONCLUSION

We have outlined a scheme for fault-tolerant universal HQC based on surface codes, with stabilizer Hamiltonian to protect quantum information encoded in the degenerate ground space, from both thermal errors and small perturbations. We explicitly constructed all necessary processes with energy gap protection and parallel operations. These processes include logical state creation, a logical universal gate set, and logical state measurement. Logical state initialization and measurement are realized by open-loop adiabatic evolution and measurements on single qubits compatible with system Hamiltonian, while the logical CNOT is implemented by a closed-loop holonomic operation. All other logical gates can be implemented using the logical CNOT, logical state preparation, and logical state measurement. It is worth mentioning that if a twist is allowed to exist on the surface, the logical Hadamard can be done much more efficiently. Conditions for active error correction are also discussed. The number of physical qubits needed to protect a logical qubit for fault-tolerant QC can reduce to the order of 10^2 , if large coupling constant J and low temperature are achievable in experiment.

Theoretical and experimental progress in non-Abelian HQC for single-qubit operations has been made recently, through both adiabatic [60] and non-adiabatic evolution [61–64] on various of physical systems. Applying our scheme to an actual physical system needs local 4-body interactions. Several theoretical proposals have been proposed to build such interactions effectively, which include low energy perturbations [54, 55, 58] of systems with strong two body interactions, and dynamic simulation [56, 57, 59]. As argued in Sec. V, the effect of such effective interaction on local error models needs further study. It is important to find out under what conditions these effective Hamiltonians behave like the ideal ones in *open* quantum systems.

We concentrated on surface codes in this paper, but we hope the methods can be extended to fault-tolerant QC schemes with constant gap protection on other topological codes, including color codes [35, 65] and Turaev-Viro codes [30].

Another interesting question is, could it be possible to do QC fault-tolerantly on an arbitrarily large scale without any active error correction? It has been shown that it is possible to do so with 6D topological color codes [66]. In our scheme on a 2D lattice, if J is very large and the temperature is sufficiently low (which is certainly a challenging engineering problem), then for practical algorithm, it may not be necessary to do active error correction. It has also been shown that a self-correcting quantum memory to store quantum information for a polynomially (or even exponentially) long time in the lattice size exists, if long range interactions between anyons is al-

lowed [67–71]. Theoretical work to realize such a long range interaction was also proposed in [59, 72]. Long range interaction can freeze the density of excited anyons on the lattice for such a long time that logical errors are quite unlikely to happen. One may ask whether such interactions can be allowed when we adiabatically deform the stabilizer Hamiltonian in our scheme. One difficulty here is that, when enlarging or moving the holes, it is hard to define the concept of anyons on the boundaries of the holes. How to introduce similar long range interactions during hole movement and enlargement is an interesting problem, and if it is possible, one may be able to implement self-correcting QC on a 2D lattice.

Addendum: When writing this manuscript, we note that Cesare, Landahl, Bacon, Flammia and Neels have published a manuscript [73] with the idea of implementing adiabatic TQC. There is a similarity of underlying spirit for both schemes: protecting quantum information with a constant energy gap during the process of quantum computation on topological codes. However, they differ a great deal in how they implement logical state preparation, measurement, state injection and uses of logical ancilla states. Also, we don't restrict ourselves to adiabatic process. Finally, we analyze the errors carefully to establish the fault-tolerance of our scheme.

ACKNOWLEDGEMENT

We would like to thank Ben Reichardt and Ching-Yi Lai for fruitful discussion of surface code and fault-tolerant quantum computation. This research was supported in part by ARO MURI Grant No. W911NF-11-0268, and by NSF Grants No. EMT-0829870 and No. TF-0830801.

Appendix A: Geometric Formulation of HQC

In this section, we introduce a more abstract geometric setting of holonomic problem which is useful to prove the results in Sec. III. We focus on the ground space for simplicity, however, the formalism is general and can be applied to any eigenspace of system Hamiltonian.

Suppose we have a family of Hamiltonians acting on the Hilbert space \mathbb{C}^N , and the ground state of each Hamiltonian is K -fold degenerate ($K < N$). The natural mathematical setting to describe this system is the principal bundle $(S_{N,K}(\mathbb{C}), G_{N,K}(\mathbb{C}), \pi, U(K))$, which consists of the Stiefel manifold $S_{N,K}(\mathbb{C})$, the Grassmann manifold $G_{N,K}(\mathbb{C})$, the projection map $\pi : S_{N,K}(\mathbb{C}) \rightarrow G_{N,K}(\mathbb{C})$, and the unitary structure group $U(K)$. We will explain the meaning of these mathematical objects in details below.

The Stiefel manifold is defined as:

$$S_{N,K}(\mathbb{C}) = \{V \in M(N, K; \mathbb{C}) | V^\dagger V = I_K\}, \quad (\text{A.1})$$

where $M(N, K; \mathbb{C})$ is the set of $N \times K$ complex matrices and I_K is the K -dimensional unit matrix. Physically, each column of $V \in S_{N,K}(\mathbb{C})$ can be regarded as a normalized

state in \mathbb{C}^N , and V can be viewed as an orthonormal set of K basis of the ground space of Hamiltonian:

$$V = \{|\varphi_1\rangle, |\varphi_2\rangle, \dots, |\varphi_K\rangle\}. \quad (\text{A.2})$$

Note that we have freedom to transfer from one orthonormal basis of to another through unitary transformation, we can define a unitary group $U(K)$ that acts on $S_{N,K}(\mathbb{C})$ from the right:

$$S_{N,K}(\mathbb{C}) \times U(K) \rightarrow S_{N,K}(\mathbb{C}), \quad (V, h) \mapsto Vh, \quad (\text{A.3})$$

by the matrix product of V and h . V and Vh can be regarded as two different orthonormal basis corresponding to the same ground space.

During the adiabatic evolution, the ground space of the Hamiltonian will change. The ground space can be represented as a K -dimensional hyperplane in \mathbb{C}^N . So we introduce the Grassmann manifold in \mathbb{C}^N :

$$G_{N,K}(\mathbb{C}) = \{P \in M(N, N; \mathbb{C}) | P^2 = P, P^\dagger = P, \text{Tr}P = K\}, \quad (\text{A.4})$$

where P is a projection operator onto the hyperplane in \mathbb{C}^N , and the condition $\text{Tr}P = K$ indicates that the dimension of the hyperplane is K . In our scenario, $P \in G_{N,K}(\mathbb{C})$ can be regarded as the projector onto the K -dimensional ground space of the Hamiltonian.

The relationship between the orthonormal basis V and ground space P can be seen as follows. We define the projection map $\pi : S_{N,K}(\mathbb{C}) \rightarrow G_{N,K}(\mathbb{C})$ as

$$\pi : V \mapsto P := VV^\dagger. \quad (\text{A.5})$$

The corresponding ground space projector can be obtained when the orthonormal basis is given. We can see that the basis V and basis Vh with $h \in U(K)$ belong to the same ground space, since

$$\pi(Vh) = (Vh)(Vh)^\dagger = Vhh^\dagger V^\dagger = VV^\dagger = \pi(V). \quad (\text{A.6})$$

For the purpose of the paper, we want to transform the ground space adiabatically during the procedure. To formulate such a process, we need also define the left action of the unitary group $U(N)$ on both $S_{N,K}(\mathbb{C})$ and $G_{N,K}(\mathbb{C})$ by the matrix product:

$$U(N) \times S_{N,K}(\mathbb{C}) \rightarrow S_{N,K}(\mathbb{C}), \quad (g, V) \mapsto gV, \quad (\text{A.7})$$

and

$$U(N) \times G_{N,K}(\mathbb{C}) \rightarrow G_{N,K}(\mathbb{C}), \quad (g, P) \mapsto gPg^\dagger. \quad (\text{A.8})$$

It is easy to check that $\pi(gV) = g\pi(V)g^\dagger$. This action is transitive: there is a $g \in U(N)$ for any $V, V' \in S_{N,K}(\mathbb{C})$ such that $V' = gV$. There is also a $g \in U(N)$ for any $P, P' \in G_{N,K}(\mathbb{C})$ such that $P' = gPg^\dagger$. So this action is sufficient to describe any ground space transformation. This is why we choose to use the form of Hamiltonian deformation in Eq. (21).

We can further study the topological structure of $S_{N,K}(\mathbb{C})$ and $G_{N,K}(\mathbb{C})$ for completeness. For each point $V \in S_{N,K}(\mathbb{C})$, we can define an isotropy group:

$$I_S(V) = \{g \in U(N) | gV = V\}, \quad (\text{A.9})$$

which is isomorphic to $U(N-K)$ for all $V \in S_{N,K}(\mathbb{C})$. Similarly, we can define an isotropy group for each $P \in G_{N,K}(\mathbb{C})$:

$$I_G(P) = \{g \in U(N) | gPg^\dagger = P\}, \quad (\text{A.10})$$

which is isomorphic to $U(K) \times U(N-K)$ for all $P \in G_{N,K}(\mathbb{C})$. Thus, $S_{N,K}(\mathbb{C}) \cong U(N)/U(N-K)$ and $G_{N,K}(\mathbb{C}) \cong U(N)/(U(K) \times U(N-K))$ [74].

The canonical connection form on $S_{N,K}(\mathbb{C})$ is defined as a $\mathfrak{u}(K)$ -valued one-form on $G_{N,K}(\mathbb{C})$:

$$A = V(P)^\dagger dV(P), \quad (\text{A.11})$$

which is a generalization of the WZ connection in Eq. (15). This is the unique connection that is invariant under the transformation in Eq. (A.3):

$$\begin{aligned} \tilde{A} &= h^\dagger V(P)^\dagger d(V(P)h) \\ &= h^\dagger Ah + h^\dagger dh. \end{aligned} \quad (\text{A.12})$$

We apply this formalism to the system dynamic of HQC. The state vector $|\psi(t)\rangle \in \mathbb{C}^N$ evolves according to the Schrödinger equation:

$$i \frac{d}{dt} |\psi(t)\rangle = H(t) |\psi(t)\rangle. \quad (\text{A.13})$$

The Hamiltonian has a spectral decomposition,

$$H(t) = \sum_{l=0}^L \varepsilon_l(t) P_l(t), \quad (\text{A.14})$$

with projection operators $P_l(t)$. Therefore, the set of energy eigenvalues $(\varepsilon_0(t), \dots, \varepsilon_L(t))$ and orthogonal projectors $(P_0(t), \dots, P_L(t))$ encodes the information of the control parameters of the system. For the ground space, we write $P_0(t)$ as $P(t)$ for simplicity. Suppose the degeneracy $K = \text{Tr}\{P(t)\}$ is constant. For all t , there exists $V(t) \in S_{N,K}(\mathbb{C})$ such that $P(t) = V(t)V^\dagger(t)$. By the adiabatic approximation, we can substitute for $|\psi(t)\rangle \in \mathbb{C}^N$ a reduced state vector $\phi(t) \in \mathbb{C}^K$:

$$|\psi(t)\rangle = V(t)\phi(t). \quad (\text{A.15})$$

Since $H(t)|\psi(t)\rangle = \varepsilon_0(t)|\psi(t)\rangle$, the Schrödinger equation (A.13) becomes

$$\frac{d\phi}{dt} + V^\dagger \frac{dV}{dt} \phi(t) = \varepsilon_0(t) \phi(t), \quad (\text{A.16})$$

and the solution can be represented formally as

$$\phi(t) = e^{-i \int_0^t \varepsilon_0(\tau) d\tau} \mathcal{P} \exp \left(- \int V^\dagger dV \right) \phi(0). \quad (\text{A.17})$$

Therefore, $\psi(t)$ can be written

$$|\psi(t)\rangle = e^{-i \int_0^t \varepsilon_0(\tau) d\tau} V(t) \mathcal{P} \exp \left(- \int V^\dagger dV \right) V^\dagger(0) |\psi(0)\rangle. \quad (\text{A.18})$$

In particular, if the system comes back to its initial point, as $P(T) = P(0)$, the holonomy $\Gamma \in U(K)$ is defined as

$$\Gamma = V^\dagger(0) V(T) \mathcal{P} \exp \left(- \int V^\dagger dV \right), \quad (\text{A.19})$$

and the final state is

$$|\psi(T)\rangle = e^{-i \int_0^T \varepsilon_0(\tau) d\tau} V(0) \Gamma \phi(0). \quad (\text{A.20})$$

According to the formula above, an operation $\Gamma \in U(K)$ is applied to the ground space.

If the condition

$$V^\dagger \cdot \frac{dV}{dt} = 0, \quad (\text{A.21})$$

is satisfied for all t , the curve $V(t)$ in $S_{N,K}(\mathbb{C})$ is called a horizontal lift of the curve $P(t) = \pi(V(t))$ in $G_{N,K}(\mathbb{C})$. Then the holonomy (A.19) is greatly simplified to

$$\Gamma = V^\dagger(0) \cdot V(T) \in U(K). \quad (\text{A.22})$$

For closed-loop HQC, given a desired unitary operation $U_{\text{op}} \in U(K)$ and a fixed initial point $P(0) \in G_{N,K}(\mathbb{C})$, we want to find a loop $P(t) \in G_{N,K}(\mathbb{C})$ with base points $P(0) = P(T)$ whose horizontal lift $V(t) \in S_{N,K}(\mathbb{C})$ produces holonomy $\Gamma = U_{\text{op}}$ according to Eq. (A.22). For open-loop adiabatic code deformation, Eq. (A.18) is general to obtain the state evolution when the adiabatic condition is satisfied.

Without loss of generality, we can always restrict ourselves to the case such that $P(t)$ has the form:

$$P(t) = U(t, 0) P(0) U^\dagger(t, 0) = U(t, 0) v_0 v_0^\dagger U^\dagger(t, 0), \quad (\text{A.23})$$

for some smooth $U(t, 0) \in U(N)$ according to Eq. (A.8). Note here, $U(t, 0)$ should be chosen such that in general, at any time t ,

$$U(t + \tau, t) P(t) U^\dagger(t + \tau, t) \neq P(t), \quad (\text{A.24})$$

for some neighborhood of t . In other word, $U(t)$ must not be in the isotropy group of $P(t)$. This condition can also stated as

$$\left[\frac{\partial}{\partial \tau} U(t + \tau, t) |_{\tau=0}, P(t) \right] \neq 0. \quad (\text{A.25})$$

The case where Eq. (A.25) equals 0 is allowed only at a finite number of points in $[0, T]$. The horizontal curve should satisfy the following set of equations:

$$\begin{aligned} V^\dagger \cdot \frac{dV}{dt} &= 0, \\ P(t) &= V(t) V^\dagger(t) = U(t, 0) v_0 v_0^\dagger U^\dagger(t, 0). \end{aligned} \quad (\text{A.26})$$

The general solution to these equations can be written as:

$$V(t) = U(t, 0) v_0 h(t, 0) \quad (\text{A.27})$$

for some $h(t, 0) \in U(K)$. Substituting Eq. (A.27) into Eq. (A.26) we get:

$$\dot{h}(t, 0) = -v_0^\dagger U^\dagger(t, 0) \dot{U}(t, 0) v_0 h(t, 0), \quad (\text{A.28})$$

which completely determines the $h(t)$, horizontal lift, and state evolution for a given adiabatic process.

Appendix B: Proof of Lemma 1, 2, 3

We first prove a lemma which will be used to prove other lemmas:

Lemma 4. $\forall g_q \in \mathcal{G}$ is in the normalizer of G_n .

Proof. For any $M \in G_n$, either $[M, Q_q] = 0$ or $\{M, Q_q\} = 0$. In the second case, we have $[Q_q, M] = 2Q_q M = 2M'$, with $M' \in G_n$.

$$\begin{aligned} g_q M g_q^\dagger &= \exp \left(i \frac{\pi}{4} Q_q \right) M \exp \left(-i \frac{\pi}{4} Q_q \right) \\ &= M + i \frac{\pi}{4} [Q_q, M] - \frac{\pi^2}{16 \cdot 2!} [Q_q, [Q_q, M]] \dots \quad (\text{B.1}) \\ &= \cos(\pi/2) M + i \sin(\pi/2) M' \\ &= i M'. \end{aligned}$$

Further, if M, Q_q are Hermitian, M' is anti-Hermitian and $g_q M g_q^\dagger$ is Hermitian. \square

1. Lemma 1

The deformation of the Hamiltonian is isospectral, so the number of logical qubits encoded in the ground space is constant, say k . The horizontal lift $V_0(t)$ for $P_0(t)$ in general can be written as $V_0(t) = U_q(t, t_{q-1}) V_0(t_{q-1}) h(t, t_{q-1})$. From Eq. (A.28), $U_q^\dagger(t, t_{q-1}) \partial_t U(t, t_{q-1}) = i \partial_t f_q(t) Q_q$,

$$\frac{\partial h}{\partial t} = i V_0^\dagger(t_{q-1}) \partial_t f_q(t) Q_q V_0(t_{q-1}) \quad (\text{B.2})$$

for $t \in [t_{q-1}, t_q]$, and

$$V_0(t_{q-1}) \partial_t h(t, 0) V_0^\dagger(t_{q-1}) = i P_0(t_{q-1}) \partial_t f_q(t) Q_q P_0(t_{q-1}). \quad (\text{B.3})$$

Since $S_j(t_0) \in G_n$ for all j , $g_l \in G_n$, for all l .

$$\begin{aligned} P_0(t_{q-1}) &= \left(\prod_{l=1}^{q-1} g_l \right) P(0) \left(\prod_{l=1}^{q-1} g_l \right)^\dagger \\ &= \left(\prod_{l=1}^{q-1} g_l \right) \prod_{j=0}^{n-k} \frac{I + S_j(0)}{2} \left(\prod_{l=1}^{q-1} g_l \right)^\dagger \quad (\text{B.4}) \\ &= \prod_{j=1}^{n-k} \frac{I + S_j(t_{q-1})}{2}, \end{aligned}$$

where $S_j(t_{q-1}) = (\prod_{l=1}^{q-1} g_l) S_j(t_0) (\prod_{l=1}^{q-1} g_l)^\dagger$ is in G_n because $\{g_l\}$ are all in the normalizer of G_n (Lemma. 4). Since $[Q_q, H(t_{q-1})] \neq 0$, so there exists at least one $S_j(t_{q-1})$ such that $\{Q_q, S_j(t_{q-1})\} = 0$. According to Eq. (B.3), $V_0(t_{q-1}) \partial_t h(t, 0) V_0^\dagger(t_{q-1}) = 0$ and $h(t, t_q) = I$. Thus $V_0(t) = U_q(t, t_{q-1}) V_0(t_{q-1})$ and $V_0(t) = U_q(t, t_{q-1}) (\prod_{l=1}^{q-1} g_l) V_0(t_0)$. From Eq. (A.18).

$$|\psi(t)\rangle = e^{-i\varepsilon_0(t-t_{q-1})} U_q(t, t_{q-1}) \left(\prod_{l=1}^{q-1} g_l \right) |\psi(t_0)\rangle. \quad (\text{B.5})$$

Setting $q = p$ and $t = t_p$, we get

$$|\psi(t_p)\rangle = e^{-i\varepsilon_0(t_p-t_{q-1})} \Omega_p |\psi(t_0)\rangle. \quad (\text{B.6})$$

2. Lemma 2

First, we show that for any $\alpha \neq \beta$, the adiabatic condition for P_{s_α} and P_{s_β} is satisfied. We have $S_j(t_l) \in G_n$ according to Lemma. 4 for $1 \leq l \leq q$. Consider the time segment $[t_q, t_{q+1}]$ first. Define the index set $\mathcal{J} = \{1, 2, \dots, n-k\}$ to be the number of terms in the Hamiltonian $H(t_q)$ with sets $\mathcal{A}_\alpha = \{j \in \mathcal{J} | \{S_j(t_q), F_\alpha\} = 0\}$, $\mathcal{B}_\alpha = \mathcal{J} \setminus \mathcal{A}_\alpha$, $\mathcal{C}_{Q_l} = \{j \in \mathcal{J} | \{S_j(t_q), Q_l\} = 0\}$ and $\mathcal{D}_{Q_l} = \mathcal{J} \setminus \mathcal{C}_{Q_l}$. Since $F_\alpha \in G_n$,

$$\begin{aligned} & F_\alpha^q P_0(t_q) (F_\alpha^q)^\dagger \\ &= \prod_{j \in \mathcal{A}_\alpha} \frac{I + S_j(t_q)}{2} \prod_{j' \in \mathcal{B}_\alpha} \frac{I - S_{j'}(t_q)}{2} \\ &= \underbrace{\prod_{m \in \mathcal{C}_{Q_{q+1}}} \frac{I + s_{\alpha_m} S_m(t_q)}{2}}_{P_{s_\alpha}^\mathcal{C}} \cdot \underbrace{\prod_{m' \in \mathcal{D}_{Q_{q+1}}} \frac{I + s_{\alpha_{m'}} S_{m'}(t_q)}{2}}_{P_{s_\alpha}^\mathcal{D}} \\ &= P_{s_\alpha}(t_q). \end{aligned} \quad (\text{B.7})$$

Here, $P_{s_\alpha}^\mathcal{C}$ and $P_{s_\alpha}^\mathcal{D}$ are short for $P_{s_\alpha}^{\mathcal{C}_{Q_{q+1}}}$ and $P_{s_\alpha}^{\mathcal{D}_{Q_{q+1}}}$. For any $\beta \neq \alpha$,

$$\begin{aligned} & P_{s_\alpha}(t) \frac{\partial H(t)}{\partial t} P_{s_\beta}(t) = \\ & -i(\partial_t f_q(t)) U_{q+1}(t, t_q) \left[P_{s_\alpha}(t_q) Q_{q+1} H(t_q) P_{s_\beta}(t_q) - \right. \\ & \left. P_{s_\alpha}(t_q) H(t_q) Q_{q+1} P_{s_\beta}(t_q) \right] U_{q+1}^\dagger(t, t_q), \end{aligned} \quad (\text{B.8})$$

where $U_{q+1}(t, t_q) = \exp(i f_{q+1}(t) Q_{q+1})$. We examine the two terms in the square brackets:

$$\begin{aligned} & P_{s_\alpha}(t_q) Q_{q+1} H(t_q) P_{s_\beta}(t_q) \\ &= \varepsilon_{s_\beta}(t_q) Q_{q+1} \prod_{m \in \mathcal{C}_{Q_{q+1}}} \frac{I - s_{\alpha_m} S_m(t_q)}{2} P_{s_\alpha}^\mathcal{D}(t_q) P_{s_\beta}^\mathcal{C}(t_q) P_{s_\beta}^\mathcal{D}(t_q), \end{aligned} \quad (\text{B.9})$$

and

$$\begin{aligned} & P_{s_\alpha}(t_q) H(t_q) Q_{q+1} P_{s_\beta}(t_q) \\ &= \varepsilon_{s_\alpha}(t_q) P_{s_\alpha}^\mathcal{C}(t_q) P_{s_\alpha}^\mathcal{D}(t_q) P_{s_\beta}^\mathcal{D}(t_q) \prod_{m \in \mathcal{C}_{Q_{q+1}}} \frac{I - s_{\beta_m} S_m(t_q)}{2} Q_{q+1}. \end{aligned} \quad (\text{B.10})$$

For those s_β such that $s_{\alpha_m} \neq s_{\beta_m}$ for any $m \in \mathcal{D}_{Q_{q+1}}$, Eq. (B.8) will be zero, and the adiabatic condition will be satisfied automatically. For those s_β such that $s_{\alpha_m} = s_{\beta_m}$ for all $m \in \mathcal{D}_{Q_{q+1}}$, it's easy to check the above two expression are not equal to zero only if $s_{\beta_m} = -s_{\alpha_m}$ for all $m \in \mathcal{C}_{Q_{q+1}}$. Therefore, there is only one β such that $P_{s_\alpha}(t) \partial_t (H(t)) P_{s_\beta}(t) \neq 0$ and hence that needs further checking. For that specific β , we have a simple relation:

$$Q_{q+1} P_{s_\alpha}(t_q) Q_{q+1}^\dagger = P_{s_\beta}(t_q), \quad (\text{B.11})$$

and

$$\begin{aligned} & \|P_{s_\alpha}(t) \frac{\partial H(t)}{\partial t} P_{s_\beta}(t)\|_1 \\ &= \partial_t f_{q+1}(t) |\varepsilon_{s_\alpha}(t_q) - \varepsilon_{s_\beta}(t_q)| \cdot \|P_{s_\alpha}(t_q) Q_{q+1}\|_1 \\ &= K \partial_t f_{q+1}(t) |\varepsilon_{s_\alpha}(t_q) - \varepsilon_{s_\beta}(t_q)|. \end{aligned} \quad (\text{B.12})$$

The left hand side of Eq. (26) reduces to

$$\frac{|\partial_t f_{q+1}(t)|}{|\varepsilon_{s_\alpha}(t_q) - \varepsilon_{s_\beta}(t_q)|}, \quad (\text{B.13})$$

since $|\mathcal{C}_{Q_{q+1}}|$ is odd. We have

$$|\varepsilon_{s_\alpha}(t_q) - \varepsilon_{s_\beta}(t_q)| = \left| \sum_{m \in \mathcal{C}_{Q_{q+1}}} 2s_{\alpha_m} \right| \geq 2. \quad (\text{B.14})$$

If $\partial_t f_{q+1}(t) \ll 1$ is satisfied (which is always possible by setting appropriate controls), then $P_{s_\alpha}(t)$ satisfies the adiabatic condition for time segment $t \in [t_q, t_{q+1}]$. The same argument can be applied to the time segments $l > q$ to show that the adiabatic condition can be satisfied between $P_{s_\beta}(t)$ and $P_{s_\alpha}(t)$ for any β . According to Eq. (A.18),

$$\begin{aligned} |\psi(t_p)\rangle &\propto V_{s_\alpha}(t_p) (F_\alpha^q V_0(t_q))^\dagger F_\alpha^q V_0(t_q) V_0^\dagger(0) |\psi(0)\rangle \\ &= V_{s_\alpha}(t_p) V_0^\dagger(t_q) V_0(t_q) V_0^\dagger(0) |\psi(0)\rangle \\ &= V_{s_\alpha}(t_p) V_0^\dagger(0) |\psi(0)\rangle, \end{aligned} \quad (\text{B.15})$$

where $V_{s_\alpha}(t)$ is defined as

$$V_{s_\alpha}(t) = U(t, t_q) F_\alpha^q V_0(t_q) h(t, t_q), \quad t > t_q, \quad (\text{B.16})$$

and is the horizontal lift of $P_{s_\alpha}(t)$ given the initial condition $F_\alpha V_0(t_q)$. From the same argument in the proof of Lemma 1, we get

$$\begin{aligned} |\psi(t_p)\rangle &= \sum_\alpha c_\alpha e^{-i\varepsilon_{s_\alpha}(t_p-t_q)} \left(\prod_{l=q+1}^p g_l \right) F_\alpha^q |\psi(t_q)\rangle \\ &= \sum_\alpha c_\alpha e^{-i\varepsilon_{s_\alpha}(t_p-t_q)} F_\alpha^{pq} \left(\prod_{l=1}^p g_l \right) |\psi(t_0)\rangle. \end{aligned} \quad (\text{B.17})$$

3. Lemma 3

For part 1, according to condition 1,

$$U_{q+1}(t, t_q) = \exp \left(i \sum_{r=q+1}^{q+M} f(t) Q_r \right), \quad (\text{B.18})$$

for $t \in [t_q, t_{q+1}]$. From the procedure in the proof of Lemma 1,

$$\begin{aligned} \frac{\partial h}{\partial t} &= i f(t) V_{s_\alpha}^\dagger(t_q) P_{s_\alpha}(t_q) \left(\sum_{Q_r \in \mathcal{P}_q} Q_r \right) P_{s_\alpha}(t_q) V_{s_\alpha}(t_q) \\ &= 0, \end{aligned} \quad (\text{B.19})$$

according to $Q_r \in G_n$ and

$$|\psi(t)\rangle = e^{-i\varepsilon_0(t-t_{q-1})} U_{q+1}(t, t_q) |\psi(t_q)\rangle. \quad (\text{B.20})$$

When $t = t_{q+1}$, when $f(t_{q+1}) = \pi/4$, and

$$|\psi(t_{q+1})\rangle = e^{-i\varepsilon_0(t_{q+1}-t_q)} \left(\prod_{l=q+1}^{q+M} g_l \right) |\psi(t_q)\rangle, \quad (\text{B.21})$$

under the adiabatic approximation.

For part 2, suppose F_α^q takes the system from the ground space to P_{s_α} . Then for any $\beta \neq \alpha$,

$$\begin{aligned} P_{s_\alpha}(t) \frac{\partial H(t)}{\partial t} P_{s_\beta}(t) &= \\ i(\partial_t f(t)) U_{q+1}(t, t_q) \sum_{Q_r \in \mathcal{P}_q} &\left[P_{s_\alpha}(t_q) Q_r H(t_q) P_{s_\beta}(t_q) - \right. \\ P_{s_\alpha}(t_q) H(t_q) Q_r P_{s_\beta}(t_q) &\left. \right] U_{q+1}^\dagger(t, t_q). \end{aligned} \quad (\text{B.22})$$

By the same argument as in the proof of Lemma 2, for each Q_r , there is only one β_r such that $P_{s_\alpha}(t_q) Q_r H(t_q) P_{s_{\beta_r}}(t_q)$ and $P_{s_\alpha}(t_q) H(t_q) Q_r P_{s_{\beta_r}}(t_q)$ do not equal 0. Since $\mathcal{C}_{Q_r} \cap \mathcal{C}_{Q_m} = \emptyset$ for any $Q_r, Q_m \in \mathcal{P}_q$, $\beta_r \neq \beta_m$ when $r \neq m$. Then, for any such β_r ,

$$\|P_{s_\alpha}(t) \frac{\partial H(t)}{\partial t} P_{s_{\beta_r}}(t)\|_1 = K \partial_t f(t) |\varepsilon_{s_\alpha}(t_q) - \varepsilon_{s_{\beta_r}}(t_q)|. \quad (\text{B.23})$$

Since $|\mathcal{C}_{Q_r}|$ is odd, then $|\varepsilon_{s_\alpha}(t_q) - \varepsilon_{s_{\beta_r}}(t_q)| \geq 2$, the adiabatic condition Eq. (26) holds for arbitrary β , and we get

$$|\psi(t_{q+1})\rangle = \sum_{\alpha} c_{\alpha} e^{-i\varepsilon_{s_\alpha}(t_{q+1}-t_q)} F_{\alpha}^{q+1,q} \left(\prod_{l=q+1}^{q+M} g_l \right) |\psi(t_q)\rangle. \quad (\text{B.24})$$

-
- [1] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2000).
 - [2] D. Aharonov and M. Ben-Or, in *Proc. 29th Annual ACM Symposium on the Theory of Computation* (ACM Press, New York, 1997) p. 176.
 - [3] D. Gottesman, *Stabilizer codes and quantum error correction*, Ph.D. thesis, California Institute of Technology (1997), eprint arXiv:quant-ph/9705052.
 - [4] D. P. DiVincenzo and P. W. Shor, *Phys. Rev. Lett.* **77**, 3260 (1996).
 - [5] E. Knill, *Nature (London)* **434**, 39 (2005).
 - [6] D. Lidar and T. Brun, *Quantum Error Correction* (Cambridge University Press, Cambridge, 2013).
 - [7] P. Zanardi and M. Rasetti, *Phys. Lett. A* **264**, 94 (1999).
 - [8] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, “Quantum computation by adiabatic evolution,” (2000), eprint arXiv:quant-ph/0001106.
 - [9] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, *Science* **292**, 472 (2001).
 - [10] A. Kitaev, *Ann. of Phys.* **303**, 2 (2003).
 - [11] M. H. Freedman, A. Kitaev, and Z. Wang, *Comm. Math. Phys.* **227**, 587 (2002).
 - [12] C. Nayak, S. H. Simon, A. Stern, M. Freedman, and S. Das Sarma, *Rev. Mod. Phys.* **80**, 1083 (2008).
 - [13] F. Wilczek and A. Zee, *Phys. Rev. Lett.* **52**, 2111 (1984).
 - [14] P. Solinas, P. Zanardi, and N. Zanghi, *Phys. Rev. A* **70**, 042316 (2004).
 - [15] M.S. Sarandy and D. A. Lidar, *Phys. Rev. A* **73**, 062101 (2006).
 - [16] P. Solinas, M. Sassetti, P. Truini, and N. Zanghi, *New. J. Phys.* **14**, 093006 (2012).
 - [17] L.-M. Duan, J. Cirac, and P. Zoller, *Science* **292**, 1695 (2001).
 - [18] Y.-C. Zheng and T. A. Brun, *Phys. Rev. A* **86**, 032323 (2012).
 - [19] J. M. Renes, A. Miyake, G. K. Brennen, and S. D. Bartlett, *New. J. of Phys.* **15**, 025020 (2013).
 - [20] O. Oreshkov, T. A. Brun, and D. A. Lidar, *Phys. Rev. Lett.* **102**, 070502 (2009).
 - [21] O. Oreshkov, T. A. Brun, and D. A. Lidar, *Phys. Rev. A* **80**, 022325 (2009).
 - [22] D. Bacon and S. T. Flammia, *Phys. Rev. Lett.* **103**, 120504 (2009).
 - [23] D. Bacon and S. T. Flammia, *Phys. Rev. A* **82**, 030303 (2010).
 - [24] Y.-C. Zheng and T. A. Brun, *Phys. Rev. A* **89**, 032317 (2014).
 - [25] T. Albash, S. Boixo, D. A. Lidar, and P. Zanardi, *New. J. Phys.* **14**, 123016 (2012).
 - [26] S.P. Jordan, E. Farhi, and P.W. Shor, *Phys. Rev. A* **74**, 052322 (2006).
 - [27] D. A. Lidar, *Phys. Rev. Lett.* **100**, 160506 (2008).
 - [28] M. S. Siu, *Phys. Rev. A* **71**, 062314 (2005).
 - [29] A. Mizel, D. A. Lidar, and M. Mitchell, *Phys. Rev. Lett.* **99**, 070502 (2007).
 - [30] R. Koenig, G. Kuperberg, and B. W. Reichardt, *Ann. of Phys.*

- 325**, 2707 (2010).
- [31] S. Bravyi, M. Hastings, and S. Michalakis, *J. Math. Phys.* **51**, 093512 (2010).
 - [32] J. R. Wootton, J. Burri, S. Iblisdir, and D. Loss, *Phys. Rev. X* **4**, 011051 (2014).
 - [33] R. Raussendorf and J. Harrington, *Phys. Rev. Lett.* **98**, 190504 (2007).
 - [34] R. Raussendorf, J. Harrington, and K. Goyal, *New J. Phys.* **9**, 199 (2007).
 - [35] A. J. Landahl, J. T. Anderson, and P. R. Rice, “Fault-tolerant quantum computing with color codes,” (2011), eprint arXiv:1108.5738.
 - [36] H. Bombin and M. Martin-Delgado, *J. Phys. A* **42**, 095302 (2009).
 - [37] H. Bombin, *New J. Phys.* **13**, 043005 (2011).
 - [38] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, *J. of Math. Phys.* **43**, 4452 (2002).
 - [39] A. G. Fowler, A. M. Stephens, and P. Groszkowski, *Phys. Rev. A* **80**, 052312 (2009).
 - [40] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, *Phys. Rev. A* **86**, 032324 (2012).
 - [41] R. Barends, J. Kelly, A. Megrant, A. Veitia, D. Sank, E. Jeffrey, T. White, J. Mutus, A. Fowler, B. Campbell, *et al.*, *Nature* **508**, 500 (2014).
 - [42] J. R. Wootton, *J. of Mod. Opt.* **59**, 1717 (2012).
 - [43] J. Edmonds, *Can. J. Math.* **17**, 449 (1965).
 - [44] S. Tanimura, D. Hayashi, and M. Nakahara, *Phys. Lett. A* **325**, 199 (2004).
 - [45] S. Tanimura, M. Nakahara, and D. Hayashi, *J. Math. Phys.* **46**, 022101 (2005).
 - [46] R. Alicki, M. Fannes, and M. Horodecki, *J. Phys. A* **42**, 065303 (2009).
 - [47] A. Messiah, *Quantum Mechanics, Vol. II* (North-Holland Publishing Co., Amsterdam, 1965).
 - [48] B. W. Reichardt, *Quant. Inf. Proc.* **4**, 251 (2005).
 - [49] S. Bravyi and A. Kitaev, *Phys. Rev. A* **71**, 022316 (2005).
 - [50] R. Raussendorf, J. Harrington, and K. Goyal, *Ann. Phys.* **321**, 2242 (2006).
 - [51] H. Bombin, *Phys. Rev. Lett.* **105**, 030403 (2010).
 - [52] G. A. Hagedorn and A. Joye, *J. Math. Anal. and Appl.* **267**, 235 (2002).
 - [53] D. A. Lidar, A. T. Rezakhani, and A. Hamma, *J. Math. Phys.* **50**, 102106 (2009).
 - [54] J. Kempe, A. Kitaev, and O. Regev, *SIAM J. Comput.* **35**, 1070 (2006).
 - [55] R. Oliveira and B. M. Terhal, *Quant. Info. Comp.* **8**, 900 (2008).
 - [56] H. Weimer, M. Müller, I. Lesanovsky, P. Zoller, and H. P. Büchler, *Nat. Phys.* **6**, 382 (2010).
 - [57] H. Weimer, M. Müller, H. Büchler, and I. Lesanovsky, *Quant. Inf. Proc.* **10**, 885 (2011).
 - [58] A. Kitaev, *Ann. of Phys.* **321**, 2 (2006).
 - [59] D. Becker, T. Tanamoto, A. Hutter, F. L. Pedrocchi, and D. Loss, *Phys. Rev. A* **87**, 042340 (2013).
 - [60] K. Toyoda, K. Uchida, A. Noguchi, S. Haze, and S. Urabe, *Phys. Rev. A* **87**, 052307 (2013).
 - [61] E. Sjöqvist, D. M. Tong, L. M. Andersson, B. Hessmo, M. Johansson, and K. Singh, *New J. Phys.* **14**, 103035 (2012).
 - [62] A. A. Abdumalikov Jr, J. Fink, Juliusson, K., M. Pechal, S. Berger, A. Wallraff, and S. Filipp, *Nature (London)* **496**, 482 (2013).
 - [63] G. Feng, G. Xu, and G. Long, *Phys. Rev. Lett.* **110**, 190501 (2013).
 - [64] C. Zu, W.-B. Wang, L. He, W.-G. Zhang, C.-Y. Dai, F. Wang, and L.-M. Duan, *Nature* **514**, 72 (2014).
 - [65] H. Bombin and M.A. Martin-Delgado, *Phys. Rev. Lett.* **97**, 180501 (2006).
 - [66] H. Bombin, R. Chhajlany, M. Horodecki, and M. Martin-Delgado, *New J. Phys.* **15**, 055023 (2013).
 - [67] A. Hamma, C. Castelnovo, and C. Chamon, *Phys. Rev. B* **79**, 245122 (2009).
 - [68] S. Chesi, B. Röthlisberger, and D. Loss, *Phys. Rev. A* **82**, 022305 (2010).
 - [69] F. L. Pedrocchi, S. Chesi, and D. Loss, *Phys. Rev. B* **83**, 115415 (2011).
 - [70] A. Hutter, J. R. Wootton, B. Röthlisberger, and D. Loss, *Phys. Rev. A* **86**, 052340 (2012).
 - [71] J. R. Wootton, *Phys. Rev. A* **88**, 062312 (2013).
 - [72] F. L. Pedrocchi, A. Hutter, J. R. Wootton, and D. Loss, *Phys. Rev. A* **88**, 062313 (2013).
 - [73] C. Cesare, A. Landahl, D. Bacon, S. Flammia, and A. Neels, “Adiabatic topological quantum computing,” (2014), eprint arXiv:1406.2690.
 - [74] M. Nakahara, *Geometry, Topology and Physics*, 2nd ed. (Institute of Physics Publishing, 2003).